

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

ANÁLISE COMPARATIVA ENTRE DOIS ALGORITMOS
QUE DETERMINAM UM CAMINHO DE MÍNIMO CUSTO EM
GRAFOS COM CUSTOS NÃO-NEGATIVOS

DISSERTAÇÃO SUBMETIDA À UNIVERSIDADE FEDERAL DE
SANTA CATARINA PARA OBTENÇÃO DO GRAU DE
MESTRE EM ENGENHARIA

CECÍLIA HARUMI IWAZAKI

FLORIANÓPOLIS

SANTA CATARINA - BRASIL

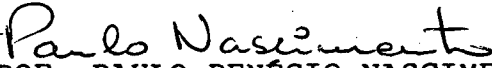
DEZEMBRO - 1987.

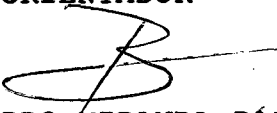
ANÁLISE COMPARATIVA ENTRE DOIS ALGORITMOS
QUE DETERMINAM UM CAMINHO DE MÍNIMO CUSTO EM
GRAFOS COM CUSTOS NÃO-NEGATIVOS

CECÍLIA HARUMI IWAZAKI

ESTA DISSERTAÇÃO FOI JULGADA ADEQUADA PARA OBTENÇÃO DO TÍTULO DE
"MESTRE EM ENGENHARIA"

ESPECIALIDADE ENGENHARIA DE PRODUÇÃO E APROVADA EM SUA
FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO


PROF. PAULO RENÉCIO NASCIMENTO, M. Sc.
ORIENTADOR


PROF. RICARDO MIRANDA BÁRCIA, Ph. D.
COORDENADOR

BANCA EXAMINADORA:


PROF. PAULO RENÉCIO NASCIMENTO, M. Sc.
PRESIDENTE


PROF. RICARDO MIRANDA BÁCIA, Ph. D.


PROF. SÉRGIO FERNANDO MAYERLE, M. Eng.

Aos meus pais

e irmãs:

Yasuko,

Sumie e

Seiko

A G R A D E C I M E N T O S

Ao Prof. PAULO RENÉCIO NASCIMENTO, pela paciência, pelo estímulo e pela excelente orientação na execução deste trabalho.

Ao Prof. ANTÔNIO DIOMÁRIO DE QUEIROZ, pelo apoio e incentivo recebido no desenvolvimento do presente trabalho.

À Universidade Federal de Santa Catarina, pelo apoio técnico e ao SENAC de Florianópolis, por ceder seu laboratório de micros, para a realização da edição inicial.

À minha irmã YASUKO IWAZAKI, pela datilografia deste trabalho.

A todos que, direta ou indiretamente, colaboraram na realização do presente trabalho.

R E S U M O

O presente trabalho tem por objetivo realizar uma análise comparativa entre dois algoritmos que determinam um caminho de mínimo custo, entre um vértice inicial e um vértice final especificados de um grafo com custos não-negativos.

Inicialmente é feito um estudo desses algoritmos, bem como suas apresentações.

Posteriormente é apresentada uma análise comparativa quanto ao desempenho computacional dos mesmos.

Finalmente são relacionados os problemas estudados e um exemplo ilustra cada procedimento.

A B S T R A C T

The objective of this work is to present a comparative analysis of two algorithms for finding the shortest path between a specified pair of nodes in a graph of nonnegative arcs.

At first, a study of these algorithms is made.

Afterwards, a comparative analysis is made concerning their computational efficiency.

Finally the studied problems are related and an example which illustrates the use of the algorithms is presented.

S U M Á R I O

CAPÍTULO I

1. PROBLEMA DE CAMINHO DE MÍNIMO CUSTO

1.1. INTRODUÇÃO.....	1
1.1.1. - GRAFOS - Um pouco de sua história.....	1
1.2. CONCEITOS BÁSICOS.....	3
1.3. PROBLEMAS DE CAMINHO DE MÍNIMO CUSTO.....	6
1.4. OBJETIVO.....	7

CAPÍTULO II

2. ALGORITMO DE DIJKSTRA

2.1. INTRODUÇÃO.....	8
2.2. DESCRIÇÃO.....	8
2.3. ALGORITMO DE DIJKSTRA - PROPRIAMENTE DITO.....	11
2.4. EXEMPLO ILUSTRATIVO DO ALGORITMO DE DIJKSTRA.....	13

CAPÍTULO III

3. ALGORITMO DE JEYARATNAM

3.1. INTRODUÇÃO.....	24
3.2. DESCRIÇÃO.....	24
3.3. ALGORITMO DE JEYARATNAM - PROPRIAMENTE DITO.....	26
3.4. EXEMPLO ILUSTRATIVO DO ALGORITMO DE JEYARATNAM.....	29

CAPÍTULO IV

4. ANÁLISE DOS RESULTADOS E CONCLUSÕES

4.1. DESCRIÇÃO DOS PROBLEMAS RESOLVIDOS.....	39
4.2. ANÁLISE DOS RESULTADOS OBTIDOS.....	39
4.3. CONCLUSÕES.....	44

BIBLIOGRAFIA.....	46
-------------------	----

ANEXO 1 - ILUSTRAÇÕES DOS PROBLEMAS RESOLVIDOS

ANEXO 2 - PROGRAMAS DOS ALGORITMOS

ANEXO 3 - RESULTADOS COMPUTACIONAIS DOS PROBLEMAS

C A P Í T U L O I

1. PROBLEMA DE CAMINHO DE MÍNIMO CUSTO

1.1. INTRODUÇÃO

O problema de busca em grafos se caracteriza pelo fato de se ter que determinar caminhos entre um nó inicial e um nó final especificados, com o menor custo possível.

A busca desses caminhos tem sido facilitada com o surgimento de vários algoritmos, desenvolvidos por diversos autores, no decorrer dos tempos, sempre com a finalidade de melhorar ainda mais o desempenho dos anteriores.

Neste sentido, o desenvolvimento da Teoria de Grafos tem participado muito na solução dos problemas de caminho de mínimo custo e com isto contribuído para resolver muitos problemas reais.

1.1.1. GRAFOS - Um pouco de sua história^{1,4}

O matemático Euler foi quem utilizou pela primeira vez os conceitos de grafos; para tal, iniciou desenvolvendo o trabalho em torno de um problema bastante conhecido: "O PROBLEMA DAS PONTES DE KONIGSBERG".

Os moradores da cidade de Königsberg tinham a seguinte situação: no rio Pregel havia 6(seis) pontes que ligavam duas ilhas às margens, além de uma sétima ponte que interligava as duas ilhas. A questão entre os frequentadores do local era que nenhum

deles poderia executar o trajeto pelas sete pontes, sem no entanto, passar mais de uma vez por uma delas.

Esse problema chegou então até os ouvidos do Matemático Euler, que logo esquematizou graficamente a situação: representou as margens (A e C) e as ilhas (B e D) por pontos (ou vértices) e as pontes por segmento de retas (ou arestas), como mostra a figura 1.

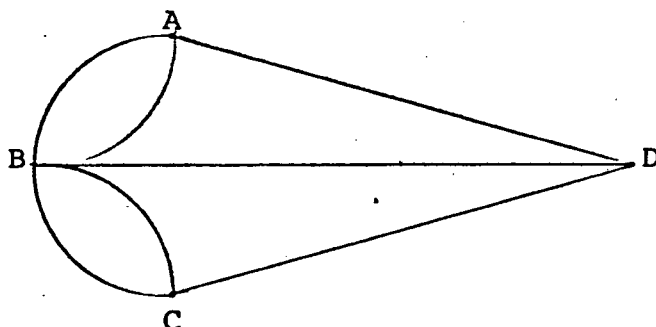


FIGURA 1: Grafo associado ao problema das Pontes de Königsberg.

Devido as disposições das pontes existentes no local, o Matemático mostrou que era impossível desenhar toda a figura sem levantar o lápis do papel, o que equivale dizer que nenhum frequentador executaria o trajeto completo passando uma e somente uma vez em cada ponte.

Várias razões e aplicações contribuíram para a aceleração do interesse na Teoria de Grafos. Frequentemente menciona-se a sua aplicação nas mais diversas áreas, cada qual recebendo uma denominação especial: "estruturas" em Engenharia Civil; "redes" em Engenharia Elétrica; "sociograma", "estrutura de comunicações" e "estrutura organizacionais" em Sociologia e Ciências Econômicas; "estrutura moleculares" em Química; "mapa rodoviário" para distribuição de gás ou eletricidade; "distribuição em redes".

1.2 - CONCEITOS BÁSICOS 2,3,4,8,11

GRAFO: Um grafo G pode ser definido como sendo uma estrutura matemática da forma $G(N,A)$, onde:

- N é um conjunto $\{n_1, n_2, \dots, n_n\}$ de elementos chamados nós ou vértices, e
- A é um conjunto $\{a_1, a_2, \dots, a_n\}$ de elementos, do produto cartesiano $N \times N$, chamados arcos ou arestas.

ARCOS: Um par ordenado de forma (i,j) é um arco, onde i é a sua extremidade inicial e j a sua extremidade final, denotados por $exi(i)$ e $ext(j)$, respectivamente.

LAÇO: Laço é um arco onde a sua extremidade inicial i coincide com a sua extremidade final j .

ANTECESSOR: O vértice i é dito antecessor de um vértice j , se existir um arco (i,j) . O conjunto de antecessores é denotado por:

$$\boxed{-1}(j) = \{i / \exists (i,j) \in A\}$$

$\boxed{-1}$ é conhecido por operador gerador de antecessores.

SUCCESSOR: O vértice j é dito sucessor de um vértice i , se existir um arco (i,j) . O conjunto de sucessores é denotado por:

$$\boxed{(1)} = \{j / \exists (i,j) \in A\}$$

$\boxed{(1)}$ é conhecido por operador gerador de sucessores.

ARESTAS: Aos arcos duplamente orientados, representados pelos pares orientados (i,j) e (j,i) , dá-se o nome

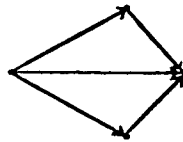
de arestas.

ARCOS OU ARESTAS ADJACENTES: São arcos ou arestas que apresentam pelo menos uma extremidade em comum, seja a extremidade inicial ou a terminal.

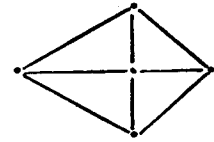
GRAFO ORIENTADO: Um grafo é dito orientado ou dirigido quando a ele estiver associado arcos (figura 2a).

GRAFO NÃO ORIENTADO: Um grafo é dito não-orientado ou não dirigido quando a ele estiver associado somente arestas (figura 2b).

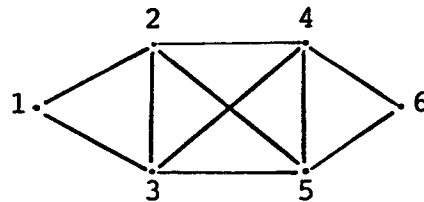
GRAFO ROTULADO: Um grafo é dito rotulado quando seus vértices estão identificados (figura 2c).



a)



b)



c)

FIGURA 2: Tipos de grafos

GRAFO SIMÉTRICO: Um grafo é dito simétrico, se e somente se para todos os arcos $(i,j) \in A \implies (j,i) \in A$.

GRAFO ANTI-SIMÉTRICO: Um grafo é dito anti-simétrico se para todos os arcos $(i,j) \in A \implies (j,i) \notin A$.

SUBGRAFO: Subgrafo de um grafo $G(N,A)$ é um grafo $G'(M,A')$,

tal que:

$$M \subset N$$

$$A' = A \cap (M \times M)$$

GRAFO PARCIAL: Grafo parcial de um grafo $G(N,A)$ é um grafo $G(N,A')$, onde:

$$A' \subset A$$

SUBGRAFO PARCIAL: Subgrafo parcial de um grafo $G(N,A)$ é um grafo $G'(N',A')$, onde:

$$N' \subset N \text{ e}$$

$$A' \subset A$$

CADEIA DE COMPRIMENTO q : Uma cadeia de comprimento q é uma sequência de q arcos $\gamma = (a_1, a_2, \dots, a_q)$ de G , tal que cada arco da sequência apresenta a $\text{ex}(a_i) = \text{ex}(a_{i+1})$, para $i = 1, 2, \dots, q-1$.

CAMINHO DE COMPRIMENTO q : Um caminho de comprimento q é uma sequência de q arcos $\gamma = (a_1, a_2, \dots, a_q)$ de G , onde a extremidade terminal de um arco, coincide com a extremidade inicial de seu sucessor, assim:

$$\text{ext}(a_i) = \text{exi}(a_{i+1}), \text{ para } i = 1, 2, \dots, q-1.$$

CAMINHO FECHADO (ou CIRCUITO): É uma sequência de q arcos $\gamma = (a_1, a_2, \dots, a_q)$ de G , onde a extremidade final da sequência coincide com a sua extremidade inicial, isto é:

$$\text{ext}(a_q) = \text{exi}(a_1)$$

CADEIA FECHADA (ou CICLO): É uma sequência de q arcos

$$\gamma = (a_1, a_2, \dots, a_q) \text{ de } G, \text{ onde:}$$

$$\text{ex}(a_1) = \text{ex}(a_q)$$

CUSTO DE UM ARCO: Considere um grafo G , onde a cada arco (i,j) , tem associado um número não negativo, denotado por $c(i,j)$, que representa o valor entre os vértices i e j , a este valor dá-se o nome de custo de (i,j) .

CUSTO DE UM CAMINHO DE COMPRIMENTO q : Dado uma sequência de q arcos $\gamma = (a_1, a_2, \dots, a_q)$ onde G é ponderado e com custos aditivos, o custo de um caminho de comprimento q é dada por:

$$C = c(a_1) + \dots + c(a_q)$$

1.3. PROBLEMA DE CAMINHO DE MÍNIMO CUSTO

Devido a sua ampla aplicabilidade, o problema de caminho de mínimo custo é considerado como o mais importante dentro do problema de busca em grafos. Como exemplo, pode-se citar as redes de distribuição que estão associadas a distâncias físicas.

Considere um grafo $G(N,A)$ com custos positivos, sendo $S \subset N$ o conjunto de nós iniciais e $T \subset N$ o conjunto de nós terminais, o problema de busca consiste em determinar um caminho de $s \in S$ para $t \in T$, com custo mínimo.

Para tal, tem-se desenvolvido vários algoritmos cuja característica básica é o uso de subgrafos até atingirem o grafo final.

Os algoritmos associados a essa busca estão classificados conforme as técnicas de pesquisa⁴, em dois processos:

- a) o processo por ajustes sucessivos, e
- b) o processo por indução e ajuste.

Os processos estão descritos a seguir:

a) PROCESSO POR AJUSTES SUCESSIVOS: seja $l(a_{ki})$ o comprimento de menor caminho entre os vértices k e i ; e λ_{ki} um custo estimado de $l(a_{ki})$. O valor de $l(a_{ki})$ é desconhecido, este então é substituído por λ_{ki} . Procura-se melhorar essa estimativa considerando os caminhos que passam pelos vértices j que sejam anteces-

sores de \underline{i} :

$$\lambda_{ki}^+ = \min [\lambda_{ki}; \lambda_{ki} + c_{ji}]$$

Essa expressão é aplicada sucessivamente aos vértices até que não seja possível ajustar nenhum valor de comprimento, quando terão sido atingidos os comprimentos dos caminhos mínimos.

b) PROCESSO POR INDUÇÃO E AJUSTE: Opera com subgrafos progressivamente maiores, até que se tenha abrangido todo o grafo; os valores vão se tornando definitivos e quando isso ocorre com todos os vértices, procura-se o itinerário correspondente ao valor mínimo encontrado.

1.4. OBJETIVO

Com o objetivo de realizar uma análise comparativa entre dois algoritmos que determinam o caminho de mínimo custo entre dois pares de nós especificados de uma rede, selecionaram-se para este trabalho os algoritmos de Dijkstra⁷ e Jeyaratnam.¹⁰

O motivo da escolha desses dois algoritmos, é que em seu artigo, Jeyaratnam afirma que, pelo menos em alguns casos, o seu algoritmo é competitivo com o de Dijkstra, considerado por vários autores como o mais eficiente de todos.

C A P Í T U L O I I

2. ALGORITMO DE DIJKSTRA

2.1. INTRODUÇÃO

Vários autores mencionam que o algoritmo de Dijkstra é considerado o mais eficiente algoritmo não heurístico para resolver os problemas de busca de caminho de mínimo custo em grafos com custos não negativos.

Para que seja possível a aplicação deste algoritmo é necessário que se tenha uma rede com N vértices, numerados arbitrariamente de 1 a N e uma matriz D , com $N \times N$ elementos, não necessariamente simétrico, cujos elementos d_{ij} representam os custos (ou distâncias) dos arcos conectando os vértices i e j .

2.2. DESCRIÇÃO^{4,5}

A técnica que o algoritmo de Dijkstra emprega para determinar o caminho de mínimo custo manipula dois conjuntos de vértices, um contendo vértices rotulados como temporários (abertos) e outro os vértices permanentes (fechados), representados por A e F , respectivamente.

Os vértices com rótulos temporários são aqueles que ainda podem ser explorados, isto é, os custos associados aos caminhos para atingí-los podem ainda ser alterados no decorrer da pesquisa. Já os vértices permanentes, não serão mais reabertos ou rotulados como temporários.

No princípio, todos os vértices são rotulados como temporários, com custos infinitos, exceto o vértice inicial que recebe o valor zero como custo de seu caminho.

Os vértices adjacentes ao inicial são atribuídos como custos, os respectivos valores dos arcos, representados na matriz D , estes passam a pertencer ao conjunto de vértices temporários.

A medida que um vértice é transferido do conjunto A para o conjunto F , determina-se os seus vértices adjacentes ou seus sucessores, bem como os custos associados de seus caminhos, adicionando-se o custo do caminho do vértice transferido com o valor do arco entre este e os seus respectivos sucessores.

Numa iteração, somente um vértice pode ser transferido do conjunto A para o conjunto F e a condição necessária é que o custo de seu caminho seja o menor entre todos da referida iteração, assim:

$$\hat{G}(k) = \min [\hat{G}(i)], \text{ para todo } i \in A \text{ e } i \notin F,$$

onde:

$\hat{G}(k)$ - é o custo mínimo do caminho entre os vértices iniciais e i ;

k - é o vértice rotulado como permanente.

Como somente um vértice por iteração recebe a rotulação permanente, pode ocorrer que dois ou mais arcos apresentem custos iguais. Mesmo neste caso é preciso selecionar apenas um e esta seleção deve ser efetuada de tal forma que a escolha não interfira no andamento eficiente da busca, nem deixa de favorecer o conjunto de nós terminais T . Quando somente um caminho de mínimo custo é desejado, pode-se fazer uma escolha arbitrária. Neste caso então, a escolha do nó mais recente pode ser considerado como

uma forma de seleção ou desempate desses arcos, sem prejudicar, no entanto, os vértices terminais.

Se um vértice já foi rotulado como permanente, este já não tem mais como ser alterado, isto quer dizer que o mesmo já foi pesquisado, porém não quer dizer que faz parte do caminho de mínimo custo, este vértice é apenas um candidato para compor o referido caminho.

Enquanto que os vértices permanentes não podem ter seus custos modificados, os temporários tem seus custos atualizados a cada iteração para que sejam testados no próximo passo.

Vários autores costumam denominar os vértices com rótulos temporários de "abertos" e os permanentes, de "fechados", neste trabalho, será usada apenas uma variável para referenciar esses dois tipos de vértices, o vetor "FEXADO". Para diferenciar se um vetor pode ser expandido ou não, foi criada uma espécie de rótulos para o vetor "FEXADO", da seguinte maneira.

$$\text{FEXADO} = \left\{ \begin{array}{l} 0 - \text{é a situação inicial de todos os vértices.} \\ 1 - \text{são os vértices possíveis de pesquisa (são} \\ \quad \text{os rotulados como temporários).} \\ 2 - \text{são os vértices já pesquisados (são os ro-} \\ \quad \text{tulados como permanentes).} \end{array} \right.$$

O vetor "FEXADO" contém apenas as informações quanto as situações dos vértices. Para reconstituir o caminho de mínimo custo foi desenvolvido um vetor apontador $P(i)$ de modo que a partir de um indicador, utilizando este sistema, tenha-se o caminho desejado.

Ao montar esse caminho, é possível determinar o seu custo, para tal, basta somar os custos dos arcos associados a estes vértices.

Para que seja possível obter todas essas informações, antes de tudo, é preciso verificar se o conjunto A é diferente de vazio, caso contrário não é possível dar continuidade a pesquisa, uma vez que não existem vértices a serem expandidos.

2.3. ALGORITMO DE DIJKSTRA - PROPRIAMENTE DITO

Para descrever as etapas do algoritmo de Dijkstra é necessário definir alguns conjuntos e algumas variáveis:

S - conjunto de vértices iniciais;

T - conjunto de vértices terminais;

V - conjunto de vértices do grafo;

$\hat{G}(i)$ - custo mínimo atual entre $s \in S$ e \underline{i} , onde \underline{i} é o vértice em análise;

$d(i, j)$ - custo do arco (distância) entre os vértices \underline{i} e \underline{j} ;

k - vértice atual rotulado como permanente;

$\lceil (k)$ - conjunto de sucessores de k ;

$P(i)$ - vetor que armazena vértices antecessores ao vértice \underline{i} ;

F e F' - custos auxiliares.

$$\text{FEXADO} = \left\{ \begin{array}{l} 0 - \text{situação inicial de todos os vértices;} \\ 1 - \text{são os vértices rotulados como temporários, representados pelo conjunto } A; \\ 2 - \text{são os vértices rotulados como permanentes, representados pelo conjunto } F. \end{array} \right.$$

O algoritmo é constituído pelas seguintes etapas:

1) Se $A = \emptyset$ então

<Pare com fracasso>

Senão

Para $i = 1, 2, \dots, n$ faça

se $FEXADO(i) = 1$ então

$$\hat{G}(k) = \min [\hat{G}(i)]^*$$

$$FEXADO(k) = 2$$

Fim se

Fim para

Fim se

$$2) A = A - \{k\}$$

$$F = F + \{k\}$$

Se $k \in T$ então

<Pare>

Senão

Gere $\lceil (k) \rceil$

Se $\lceil (k) \rceil = \emptyset$ então

Volte para 1)

Fim se

Fim se

3) Para cada arco (k, m) , $m \in \lceil (k) \rceil$, faça

$$F' = \hat{G}(k) + d(k, m)$$

Se $(m \in A \text{ e } \hat{G}(m) < F')$ ou $m \in F$ então

Volte para 3)

Senão

$$F = F'$$

$$\hat{G}(k) = F$$

$$P(m) = k$$

$$FEXADO(m) = 1$$

$$A = A \cup \{ m \}$$

Fim se

Fim para

4) Volte para 1)

* Caso exista dois ou mais caminhos com custos iguais, seleciona-se o vértice tal que beneficie o conjunto T de vértices terminais.

OBS: Para a recuperação do caminho de mínimo custo, vide programa - ANEXO 2.

2.4. EXEMPLO ILUSTRATIVO DO ALGORITMO DE DIJKSTRA

A figura 3 apresenta uma rede com 10 vértices, onde os custos (ou distâncias) dos arcos estão indicados na própria configuração.

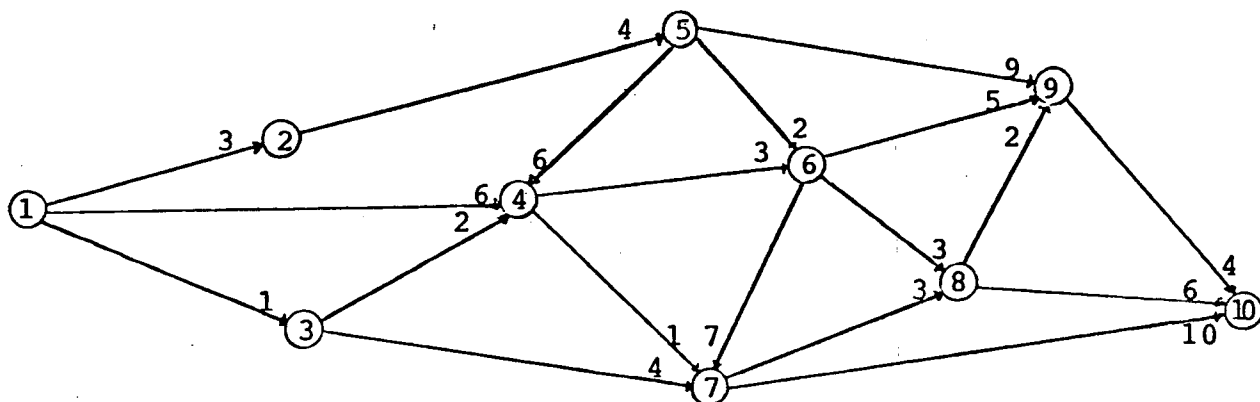


FIGURA 3.

Inicialmente:

$$N = 10$$

$$V = \{ 1, 2, 3, \dots, 10 \}$$

$$S = \{1\}$$

$$T = \{10\}$$

$$A = \{1\}$$

$$F = \emptyset$$

$$k = 1 \in V$$

$$\hat{G}(k) = 0$$

$$P(k) = 0$$

NOTA: Para identificar os vértices rotulados como permanentes, estes estão assinalados por \otimes .

ITERAÇÃO 1:

Nesta primeira iteração apresenta os seguintes resultados:

$$A = \{1\} \neq \emptyset$$

então
$$\hat{G}(k) = \min [\hat{G}(1)] = \min [0]$$

portanto
$$k = 1$$

$$FEXADO(1) = 2$$

$$A = \emptyset$$

$$F = \{1\}$$

Como $1 \notin T$, gera-se então seus sucessores e determina-se os respectivos custos (figura 3.a).

$$\overline{N}(1) = \{2, 3, 4\}$$

$$\hat{G}(2) = 3; P(2) = 1; A = \{2\}$$

$$\hat{G}(3) = 1; P(3) = 1; A = \{2, 3\}$$

$$\hat{G}(4) = 6; P(4) = 1; A = \{2, 3, 4\}$$

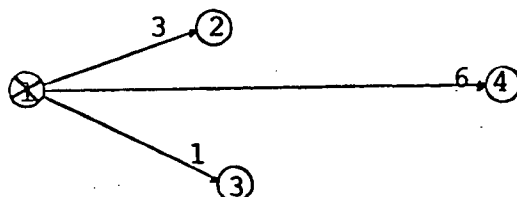


FIGURA 3.a

ITERAÇÃO 2:

Para esta iteração, o conjunto de vértices temporários apresenta 3 elementos:

$$A = \{ 2, 3, 4 \} \neq \emptyset$$

então $\hat{G}(k) = \min [\hat{G}(2), \hat{G}(3), \hat{G}(4)]$

$$\hat{G}(k) = \min [3, 1, 6]$$

portanto $k = 3$

$$\text{FEXADO}(3) = 2$$

$$A = \{ 2, 3 \}$$

$$F = \{ 1, 3 \}$$

Como $3 \notin T$, gera-se então seus sucessores e determina-se seus custos:

$$\Gamma(3) = \{ 4, 7 \}$$

$$\hat{G}(4) = 3; P(4) = 3; A = \{ 2, 4 \}$$

$$\hat{G}(7) = 5; P(7) = 3; A = \{ 2, 4, 7 \}$$

O custo do caminho até o vértice 4 foi alterado pois o custo a partir do nó 3 é melhor que o custo através do nó 1 (figura 3.b).

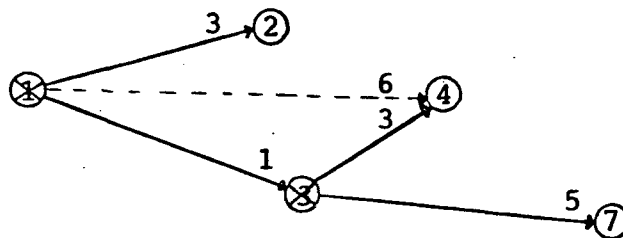


FIGURA 3.b

ITERAÇÃO 3:

A situação encontrada nesta iteração é a seguinte:

$$A = \{ 2, 4, 7 \} \neq \emptyset$$

então $\hat{G}(k) = \min [\hat{G}(2), \hat{G}(4), \hat{G}(7)]$

$$\hat{G}(k) = \min [3, 3, 5]$$

portanto

$$k = 4$$

$$\text{FEXADO}(4) = 2$$

$$A = \{ 2, 7 \}$$

$$F = \{ 1, 3, 4 \}$$

Nesta iteração os vértices 2 e 4 apresentaram custos iguais, como é preciso optar por apenas um, o nó 4 foi o selecionado, de acordo com a regra já estabelecida.

Como 4 \in T, então determina-se seus sucessores bem como os seus custos.

$$\Gamma(4) = \{ 6, 7 \}$$

$$\hat{G}(6) = 6; P(6) = 4; A = \{ 2, 6, 7 \}$$

$$\hat{G}(7) = 4; P(7) = 4; A = \{ 2, 6, 7 \}$$

O custo do caminho até o vértice 7 foi alterado, devido ao fato de que o caminho através do nó 4 é melhor que o do nó 3 (figura 3.c).

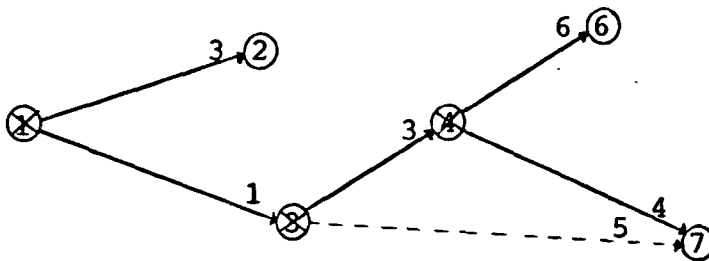


FIGURA 3.c

ITERAÇÃO 4:

Os resultados para esta iteração são os seguintes:

$$A = \{ 2, 7, 6 \} \neq \emptyset$$

então

$$\hat{G}(k) = \min [\hat{G}(2), \hat{G}(7), \hat{G}(6)]$$

$$\hat{G}(k) = \min [3, 4, 6]$$

portanto

$$k = 2$$

$$\text{FEXADO}(2) = 2$$

$$A = \{ 7, 6 \}$$

$$F = \{ 1, 3, 4, 2 \}$$

O vértice 2 $\notin T$, determina-se então seus sucessores e seus custos (figura 3.d).

$$\Gamma(2) = \{ 5 \}$$

$$\hat{G}(5) = 7; P(5) = 2; A = \{ 7, 6, 5 \}$$

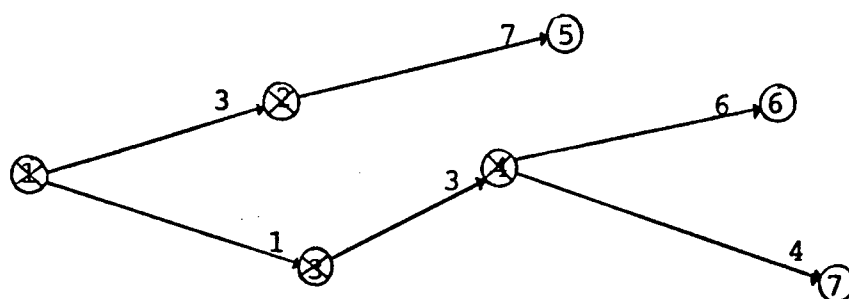


FIGURA 3.d

ITERAÇÃO 5:

Para esta iteração tem-se os seguintes resultados:

$$A = \{ 7, 6, 5 \} \neq \emptyset$$

então

$$\hat{G}(k) = \min [\hat{G}(7), \hat{G}(6), \hat{G}(5)]$$

$$\hat{G}(k) = \min [4, 6, 7]$$

portanto

$$k = 7$$

$$\text{FEXADO}(7) = 2$$

$$A = \{ 6, 5 \}$$

$$F = \{ 1, 3, 2, 4, 7 \}$$

Como 7 $\in T$, determina-se os seus sucessores e os respectivos custos (figura 3.e).

$$\Gamma(7) = \{ 8, 10 \}$$

$$\hat{G}(8) = 7; P(8) = 7; A = \{ 6, 5, 8 \}$$

$$\hat{G}(10) = 14; P(10) = 7; A = \{ 6, 5, 8, 10 \}$$

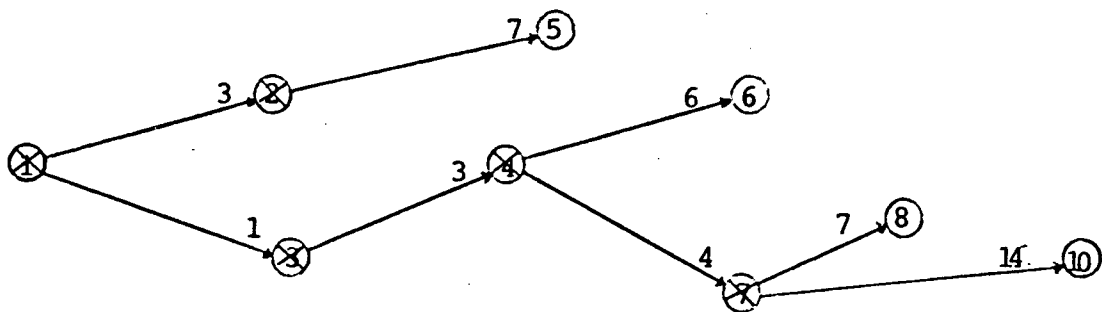


FIGURA 3.e

ITERAÇÃO 6:

Para esta iteração a situação é a seguinte:

$$A = \{ 6, 5, 8, 10 \} \neq \emptyset$$

então $\hat{G}(k) = \min [\hat{G}(6), \hat{G}(5), \hat{G}(8), \hat{G}(10)]$

$$\hat{G}(k) = \min [6, 7, 7, 14]$$

portanto $k = 6$

$$FEXADO(6) = 2$$

$$A = \{ 5, 8, 10 \}$$

$$F = \{ 1, 3, 2, 4, 7, 6 \}$$

$6 \notin T$, então calcula-se os custos de seus sucessores e determina-se os mesmos (figura 3.f).

$$T(6) = \{ 7, 8, 9 \}$$

$$\hat{G}(9) = 11; P(9) = 6; A = \{ 5, 8, 10, 9 \}$$

Como $FEXADO(7) = 2$, este já não é mais reaberto e o custo do caminho para o vértice 8 é melhor aquele através do no 7 este é então preferido.

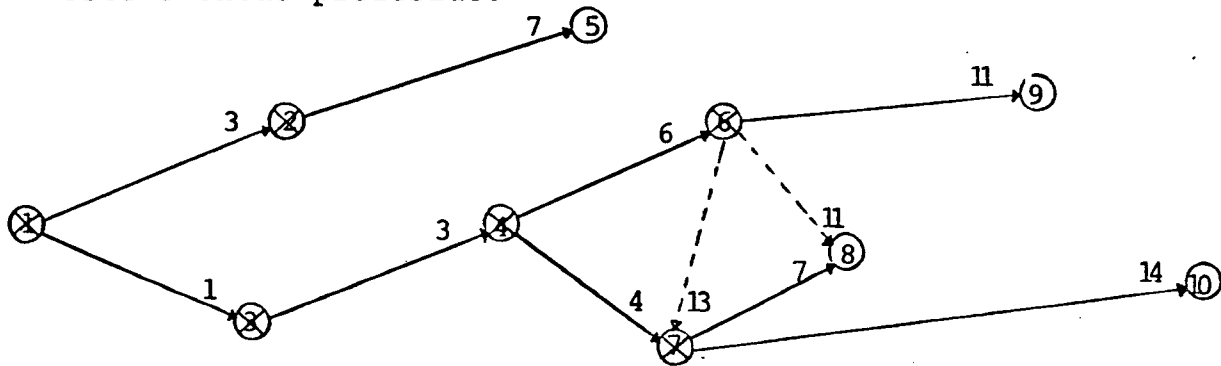


FIGURA 3.f

ITERAÇÃO 7:

Nesta iteração a situação da pesquisa é a seguinte:

$$A = \{ 5, 8, 10, 9 \} \neq \emptyset$$

então $\hat{G}(k) = \min [\hat{G}(5), \hat{G}(8), \hat{G}(10), \hat{G}(9)]$

$$\hat{G}(k) = \min [7, 7, 14, 11]$$

portanto $k = 8$

$$FEXADO(8) = 2$$

$$A = \{ 5, 10, 9 \}$$

$$F = \{ 1, 3, 2, 4, 7, 6, 8 \}$$

Novamente dois caminhos apresentaram custos iguais, optou-se pelo nó 8, porém $8 \notin T$, então gera-se seus sucessores e determina-se os custos dos mesmos (figura 3.g).

$$T(8) = \{ 9, 10 \}$$

$$\hat{G}(9) = 9; P(9) = 8; A = \{ 5, 10, 9 \}$$

$$\hat{G}(10) = 13; P(10) = 8; A = \{ 5, 10, 9 \}$$

Ambos os custos desses sucessores foram modificados, pois estes são melhores que os anteriores.

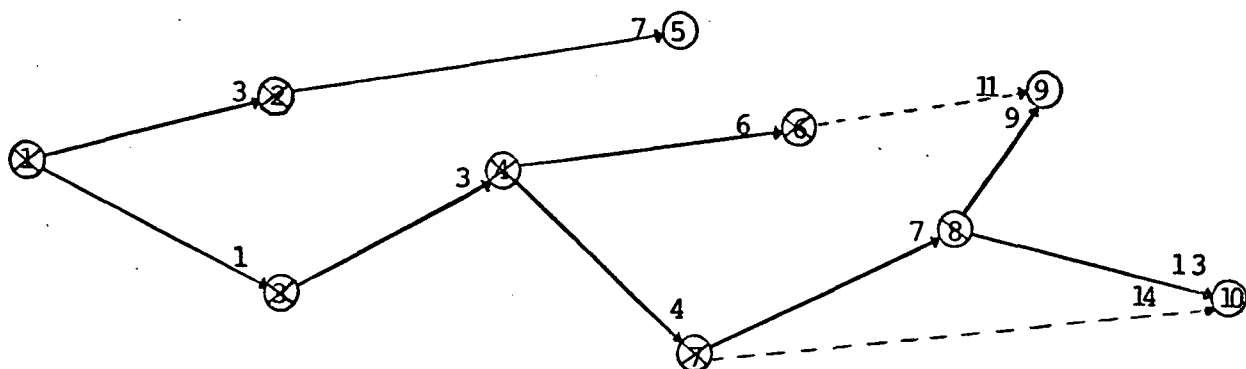


FIGURA 3.g

ITERAÇÃO 8:

Para esta iteração, os resultados são os seguintes:

$$A = \{ 5, 10, 9 \} \neq \emptyset$$

então $\hat{G}(k) = \min [\hat{G}(5), \hat{G}(10), \hat{G}(9)]$

$$\hat{G}(k) = \min [7, 13, 9]$$

portanto

$$k = 5$$

$$\text{FEXADO}(5) = 2$$

$$A = \{ 10, 9 \}$$

$$F = \{ 1, 3, 2, 4, 7, 6, 8, 5 \}$$

Como $5 \notin T$, gera-se então os seus sucessores e determina-se os custos de seus caminhos (figura 3.h).

$$\Gamma(5) = \{ 4, 6, 9 \}$$

Como $\text{FEXADO}(6) = 2$ e $\text{FEXADO}(4) = 2$, estes vértices já não são mais reabertos. Quanto ao custo do caminho para o vértice 9, já existe um outro melhor, através do vértice 8.

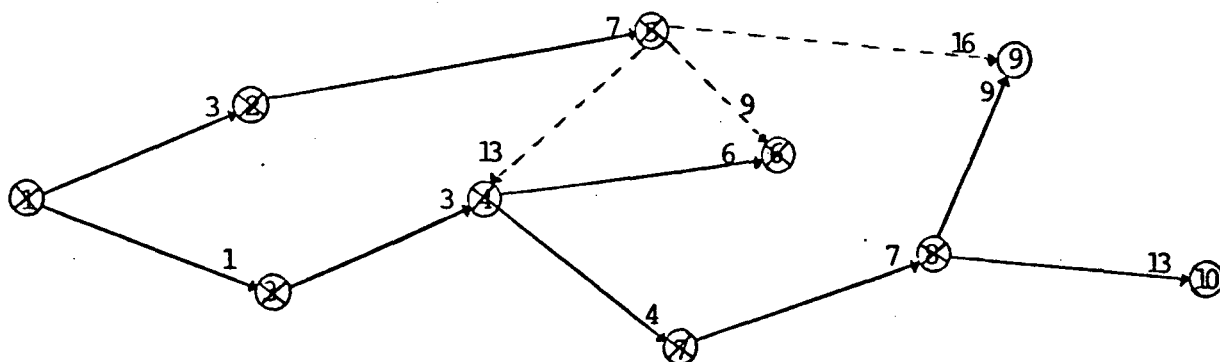


FIGURA 3.h

ITERAÇÃO 9:

Os resultados para esta iteração são os seguintes:

$$A = \{ 10, 9 \} \neq \emptyset$$

então

$$\hat{G}(k) = \min [\hat{G}(10), \hat{G}(9)]$$

$$\hat{G}(k) = \min [13, 9]$$

portanto

$$k = 9$$

$$\text{FEXADO}(9) = 2$$

$$A = \{ 10 \}$$

$$F = \{ 1, 3, 2, 4, 7, 6, 8, 5, 9 \}$$

$9 \notin T$, gera-se então os seus sucessores e determinam-se os custos desses caminhos (figura 3.i).

$$\begin{aligned} \Gamma(9) &= \{ 10 \} \\ \hat{G}(10) &= 13; P(10) = 9; A = \{ 10 \} \end{aligned}$$

Como o caminho para o vértice 10 é melhor através do vértice 9, este é então preferido ao anterior.

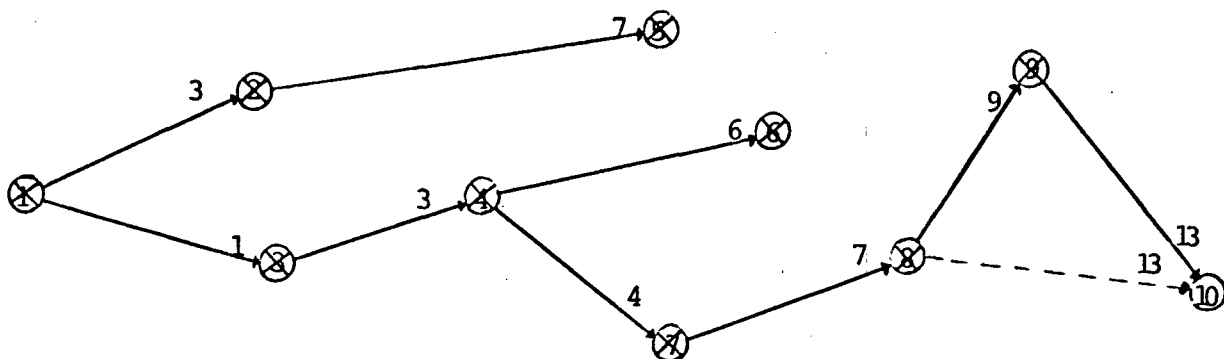


FIGURA 3.i

ITERAÇÃO 10:

Para esta iteração, os resultados são os seguintes:

$$A = \{ 10 \} \neq \emptyset$$

então

$$\hat{G}(k) = \min [\hat{G}(10)] = \min [13]$$

portanto

$$k = 10$$

$$\text{FEIXADO}(10) = 2$$

$$A = \emptyset$$

$$F = \{ 1, 3, 2, 4, 7, 6, 8, 5, 9, 10 \}$$

Como $10 \in T$, o algoritmo se encerra com um custo mínimo igual a 13 e com o seguinte caminho:

$$10, 9, 8, 7, 4, 3, 1$$

Vide figura 3.j.

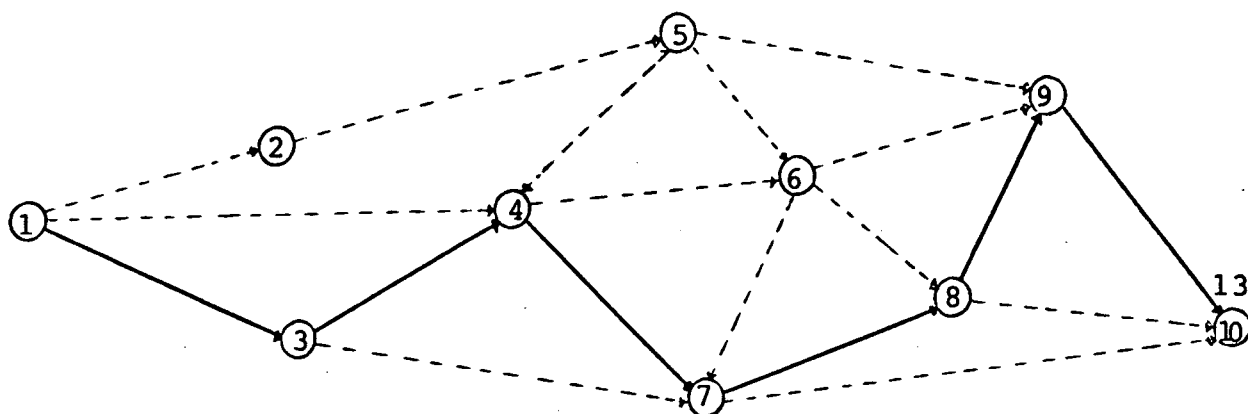


FIGURA 3.j

$$C = 13$$

$$\gamma = (1, 3, 4, 7, 8, 9, 10)$$

COMETÁRIOS:

- o referido algoritmo realiza a pesquisa apenas em um sentido do início ao final, rotulando os vértices como temporários ou permanentes;

- conseqüentemente, utiliza apenas um sistema de apontador para a reconstituição do caminho;

- e esta reconstituição é efetuada a partir do último vértice, rotulado como permanente, até o vértice inicial $s \in S$.

C A P Í T U L O I I I

3. ALGORITMO DE JEYARATNAM¹⁰

3.1. INTRODUÇÃO

Este novo algoritmo, proposto por Jeyaratnam, para determinar o caminho de mínimo custo entre dois vértices especificados de uma rede com custos não negativos, segundo o autor, torna-se competitivo, pelo menos em alguns casos, com o algoritmo de Dijkstra, descrito no capítulo anterior.

Para que seja possível aplicar este algoritmo é necessário que as redes apresentem as características já requeridas pelo algoritmo de Dijkstra.

3.2. DESCRIÇÃO

O método empregado neste algoritmo para resolver o problema de busca do caminho de mínimo custo em grafos, também consiste em rotular os vértices como temporários ou permanentes, porém, as pesquisas são efetuadas em dois sentidos, do início para o final, e do final para o início.

Esta técnica se assemelha com o procedimento apresentado por Nicholson,¹² que pesquisa simultaneamente em ambos os sentidos e que após encerrado o processo de rotulação reconstitui o caminho dinamicamente.

Este algoritmo apresenta o seguinte desenvolvimento:

inicialmente, todos os vértices são rotulados como temporários, exceto os nós inicial e final que já iniciam como permanentes. Aos arcos associados aos vértices inicial e final, são atribuídos custos nulos permanentemente. Os arcos associados a outros vértices, que apresentam conexão com esses dois, tem seus custos modificados pelos respectivos valores de seus arcos, representados na matriz D e ao demais, são alocados custos bem grandes.

Estas duas formas de identificação dos vértices são manipulados por um vetor denominado "FEXADO", que através de um número, indica se os mesmos são temporários ou permanentes.

Neste algoritmo, as mudanças de rótulos são efetuadas alternadamente com o mesmo critério já descrito anteriormente, ou seja, é rotulado como permanente o vértice, que na iteração, estiver com o seu arco associado valorado com o menor custo de todos, não importando o sentido da busca.

A cada iteração, após as devidas rotulações, os custos dos vértices rotulados como temporários são atualizados, quando possível.

Como todas as operações são realizadas alternadamente, fica caracterizado que este algoritmo apresente dois passos fundamentais, um que realiza a busca do início para o final e o outro que a realiza do final para o início, sem no entanto perder as informações já obtidas, devidamente armazenadas e transportadas para as iterações seguintes.

Independente do sentido da pesquisa, apenas um vértice pode ser rotulado como permanente. No caso em que dois ou mais caminhos apresentam custos iguais, seleciona-se o vértice com o menor índice. Esta é uma opção arbitrária, uma vez que se deseja

APENAS UM caminho de mínimo custo.

Em termos computacionais, para armazenar as informações a respeito dos vértices, utilizou-se também o vetor "FEXADO", cuja função é a mesma apresentada no algoritmo de Dijkstra. Embora as rotulações sejam efetuadas em ambos os sentidos, foi utilizado apenas este vetor, devido ao fato de que todas as informações são transferidas de um passo para o outro.

Como neste algoritmo a busca é efetuada em ambos os sentidos, a reconstituição do caminho também foi realizada em duas etapas, desenvolvendo dois sistemas de apontadores:

- o vetor PAI, cuja função é armazenar os vértices permanentes pesquisados do início para o final; e
- o vetor FILHO, cuja função é armazenar os vértices permanentes pesquisados do final para o início.

Além desses vetores foi preciso definir um ponto de partida, denominado de M, que é o ponto de encontro entre os vetores PAI e FILHO. O objetivo deste ponto é para orientar na reconstituição do caminho de mínimo custo da seguinte maneira:

- através do vetor PAI é tomado como base o ponto M, forma-se uma parte do caminho de forma retroativa, que, devidamente ordenada, fica armazenada em um vetor auxiliar. O vetor FILHO, tendo como ponto de partida o ponto M, é utilizado na formação da segunda parte do caminho, que anexada a primeira, determina o caminho completo desejado.

O custo associado a esse caminho fica determinado na medida em que se desenvolve a busca, com o valor sendo armazenado na variável MIN. Este valor além de representar o custo mínimo do caminho, é utilizado também na finalização do algoritmo.

Este algoritmo apresenta três regras de parada:

- quando todos os vértices já foram pesquisados e rotulados como permanentes;
- quando o valor da variável MIN for menor ou igual a soma dos custos do vértice inicial ao vértice em questão;
- e
- quando o valor da variável MIN for menor ou igual a soma dos custos do vértice final ao vértice em questão.

3.3. ALGORITMO DE JEYARATNAM - PROPRIAMENTE DITO

Para descrever os dois passos e as etapas do algoritmo de Jeyaratnam é preciso definir alguns conjuntos e algumas variáveis:

S - conjunto de vértices (nós) iniciais;

T - conjunto de vértices (nós) terminais;

V - conjunto de vértices do grafo;

$\hat{G}(i)$ - custo mínimo atual entre $s \in S$ e i , onde i é o vértice em análise;

$\hat{G}'(i)$ - custo mínimo atual entre $t \in T$ e i , onde i é o vértice em análise;

G(i) - custo temporário atual entre $s \in S$ e i , onde i é o vértice em análise;

G'(i) - custo temporário atual entre $t \in T$ e i , onde i é o vértice em análise;

d(i,j) - custo (ou distância) entre os vértices i e j ;

k - vértice atual rotulado como permanente para a pesquisa realizada do início para o final;

u - vértice atual rotulado como permanente para a pesquisa realizada do final para o início;

PAI - vetor que armazena o vértice antecessor para a pesquisa realizada do início para o final;

FILHO - vetor que armazena o vértice antecessor quando a pesquisa se realiza do final para o início;

FEXADO= $\left\{ \begin{array}{l} 0 - \text{situação inicial de todos os vértices;} \\ 1 - \text{são os vértices rotulados como temporários,} \\ \quad \text{representados pelo conjunto A;} \\ 2 - \text{são os vértices rotulados como permanentes,} \\ \quad \text{representados pelo conjunto F;} \end{array} \right.$

NF - número de vértice com rótulos permanentes, ou seja, quando FEXADO=2.

PASSO 1:

Para cada arco (k,i) , onde $FEXADO(i) \neq 2$, faça

Se $G(i) > \hat{G}(k) + d(k,i)$ então

$G(i) = \hat{G}(k) + d(k,i)$

$PAI(i) = k$

$FEXADO(i) = 1$

Fim se

Fim para

Para todo i , onde $FEXADO(i) = 1$, faça

$G(k) = \min [G(i)]^*$

Fim para

Se $MIN. < = G(k)$ então

<FIM>

Senão

$k = K$

$\hat{G}(k) = G(K)$

$FEXADO(k) = 2$

$$F = F \cup \{k\}$$

$$NF = NF + 1$$

Fim se

Se $G'(k) > \hat{G}'(u) + d(k,u)$ então

$$G'(k) = \hat{G}'(u) + d(k,u)$$

$$FILHO(k) = u$$

Fim se

Se $MIN > \hat{G}(k) + G'(k)$ então

$$MIN = \hat{G}(k) + G'(k)$$

$$M = k$$

Fim se

Se $NF = N$ então

<FIM>

Senão

Vá para o PASSO 2)

Fim se

PASSO 2:

Para cada arco (j,u) , onde $FEXADO(j) \neq 2$, faça

Se $G'(j) > G'(u) + d(j,u)$ então

$$G'(j) = G'(u) + d(j,u)$$

$$FILHO(j) = u$$

$$FEXADO(j) = 1$$

Fim se

Fim para

Para todo j , onde $FEXADO(j) = 1$, faça

$$G'(U) = \text{Min} [G'(j)]^*$$

Fim para

Se $MIN \leq G'(U)$ então

<FIM>

Senão

$$u = U$$

$$\tilde{G}'(u) = G'(U)$$

$$\text{FEXADO}(u) = 2$$

$$F = F \cup \{ u \}$$

$$NF = NF + 1$$

Fim se

Se $G(u) > \tilde{G}(k) + d(k,u)$, então

$$G(u) = \tilde{G}(k) + d(k,u)$$

$$PAI(u) = k$$

Fim se

Se $MIN > G(u) + \tilde{G}'(u)$ então

$$MIN = G(u) + \tilde{G}'(u)$$

$$M = u$$

Fim se

Se $NF = N$ então

<FIM>

Senão

Volte para o PASSO 1)

Fim se

- * Caso exista dois ou mais caminhos com custos iguais, seleciona-se o vértice de menor número.

3.4. EXEMPLO ILUSTRATIVO DO ALGORITMO DE JEYARATNAM

Para este exemplo, considere a mesma rede apresentada na

figura 3, do capítulo anterior, com as seguintes condições iniciais:

$$N = 10$$

$$S = \{ 1 \}$$

$$T = \{ 10 \}$$

$$\sum_{i=1,n}^{j=1,n} d_{ij} = 75$$

$$k = 1$$

$$t = 10$$

$$\text{MIN} = 75$$

$$\text{FEXADO}(k) = 2$$

$$\text{FEXADO}(t) = 2$$

$$\hat{G}(k) = 0$$

$$\hat{G}(t) = 0$$

$$F = \{ 1, 10 \}$$

$$NF = 2$$

$$G(i) = d_{1,j} \quad , \text{ para } i = 2, 3, 4, \dots, n$$

$$G'(j) = d_{j,n} \quad , \text{ para } j = 1, 2, 3, \dots, n-1$$

$$\text{PAI}(i) = 1 \quad , \text{ para } i = 1, 2, 3, \dots, n$$

$$\text{FILHO}(i) = 10 \quad , \text{ para } i = 1, 2, 3, \dots, n$$

$$M = 1$$

NOTA 1: Convencionou-se que as iterações ímpares são para as pesquisas realizadas do início para o final e as iterações pares, para as pesquisas realizadas do final para o início.

NOTA 2: Os vértices assinalados por \otimes são os rotulados como permanentes.

ITERAÇÕES 1 E 2:

Para a iteração 1, os resultados são os seguintes:

$G(2) = 3; PAI(2) = 1$
 $G(3) = 1; PAI(3) = 1$
 $G(4) = 4; PAI(4) = 1$
 então $G(k) = \min [3, 1, 6]$
 $k = 3$
 como $MIN = 75 > G(3)$
 então $k = 3$
 $\hat{G}(3) = 1$
 $FEXADO(3) = 2$
 $F = \{ 1, 10, 3 \}$
 $NF = 3$

Como $NF \neq 10$, a busca do caminho continua.

Para a iteração 2, os resultados são os seguintes:

$G'(7) = 10; FILHO(7) = 10$
 $G'(8) = 6; FILHO(8) = 10$
 $G'(9) = 4; FILHO(9) = 10$
 então $G'(U) = \min [10, 6, 4]$
 $U = 9$
 como $MIN = 75 > G'(9)$
 então $u = 9$
 $\hat{G}(9) = 4$
 $FEXADO(9) = 2$
 $F = \{ 1, 10, 3, 9 \}$
 $NF = 4$

Como $NF \neq 10$, a busca continua. Os resultados destas duas iterações estão ilustrados na figura 4.a).

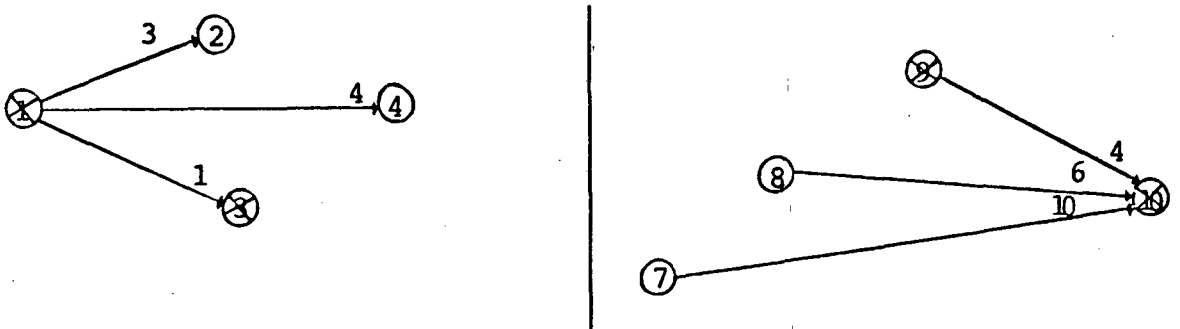


FIGURA 4.a

ITERAÇÕES 3 E 4:

Os resultados para a iteração 3 são:

$$G(4) = 3; \text{ PAI}(4) = 3$$

$$G(7) = 5; \text{ PAI}(7) = 3 \text{ e}$$

$$G(2) = 3; \text{ PAI}(2) = 1, \text{ da iteração 1.}$$

O custo do caminho para o vértice 4 foi alterado, pois o caminho através do nó 3 é melhor que o do nó 1.

então $G(k) = \min [3, 3, 5]$

$$k = 2$$

$$\text{MIN} = 75 > G(2)$$

portanto

$$k = 2$$

$$\tilde{G}(2) = 3$$

$$\text{FEXADO}(2) = 2$$

$$F = \{ 1, 10, 3, 9, 2 \}$$

$$\text{NF} = 5$$

Nesta iteração, dois caminhos apresentaram custos iguais, optou-se por aquele com o menor número, como foi descrito anteriormente como regra de seleção.

Como $\text{NF} \neq 10$, a busca continua.

Os resultados para a iteração 5, são os seguintes:

$$G'(5) = 13; \text{FILHO}(5) = 9$$

$$G'(6) = 9; \text{FILHO}(6) = 9$$

$$G'(8) = 6; \text{FILHO}(8) = 9 \text{ e}$$

$$G'(7) = 10; \text{FILHO}(7) = 10 \text{ da iteração 2.}$$

O custo do caminho para o vértice 8 foi alterado, pois o custo através do vértice 9 é melhor que o anterior.

então $G'(U) = \min [13, 9, 10, 6]$

$$U = 8$$

$$\text{MIN} = 75 > G'(8)$$

portanto $u = 8$

$$\hat{G}'(8) = 6$$

$$\text{FECHADO}(8) = 2$$

$$F = \{ 1, 10, 3, 9, 2, 8 \}$$

$$NF = 6$$

$NF \neq 10$, a busca do caminho continua e os resultados destas iterações estão ilustradas na figura 4.b.

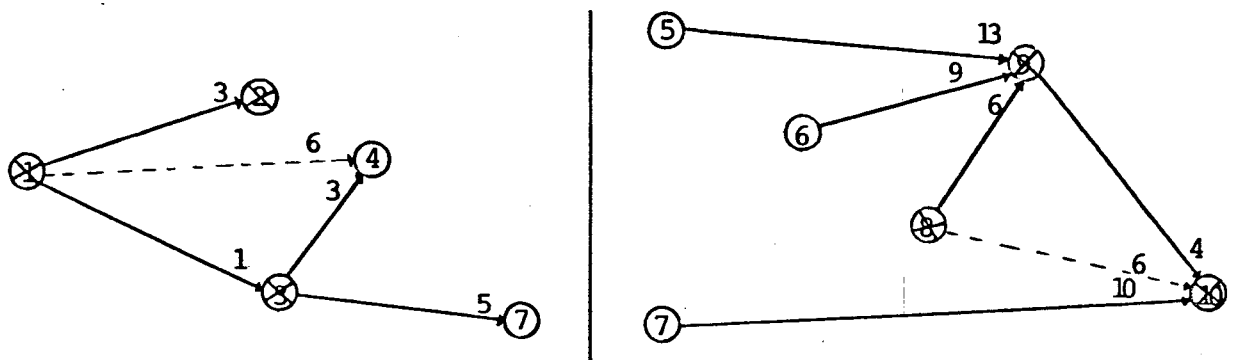


FIGURA 4.b

ITERAÇÕES 5 E 6:

Para a iteração 5, os resultados são os seguintes:

$$G(5) = 7; \text{ PAI}(5) = 2 \text{ e}$$

$$G(4) = 3; \text{ PAI}(4) = 3 \text{ da iteração 3 e}$$

$$G(7) = 5; \text{ PAI}(7) = 3 \text{ da iteração 3.}$$

então

$$G(k) = \min [3, 7, 5]$$

$$k = 4$$

$$\text{MIN} = 75 > G(4)$$

portanto

$$k = 4$$

$$\hat{G}(4) = 3$$

$$\text{FEXADO}(4) = 2$$

$$F = \{ 1, 10, 3, 9, 2, 8, 4 \}$$

$$\text{NF} = 7$$

Como $\text{NF} \neq 10$, a busca ainda continua.

Para a iteração 6, tem-se os seguintes resultados:

$$G'(6) = 9; \text{ FILHO}(6) = 8$$

$$G'(7) = 9; \text{ FILHO}(7) = 8 \text{ e}$$

$$G'(5) = 13; \text{ FILHO}(5) = 9 \text{ da iteração 4.}$$

Os custos dos caminhos para os vértices 6 e 7 foram alterados, estes custos são melhores que os anteriores.

então

$$G'(U) = \min [13, 9, 9]$$

$$U = 6$$

$$\text{MIN} = 75 > G'(6)$$

portanto

$$u = 6$$

$$\hat{G}'(6) = 9$$

$$\text{FEXADO}(6) = 2$$

$$F = \{ 1, 10, 3, 9, 2, 8, 4, 6 \}$$

$$\text{NF} = 8$$

como

$$G(6) > G(4) + d(4, 6)$$

então $G(6) = 6$ e
 $PAI(6) = 4$
 como $MIN > G(6) + G'(6)$
 então $MIN = 15$
 $M = 6$

Nesta iteração dois caminhos apresentaram custos iguais, para selecionar um deles, utilizou-se da regra já estabelecida.

Como $NF \neq 10$, a pesquisa ainda continua e os resultados destas duas iterações, estão ilustrados na figura 4.c.

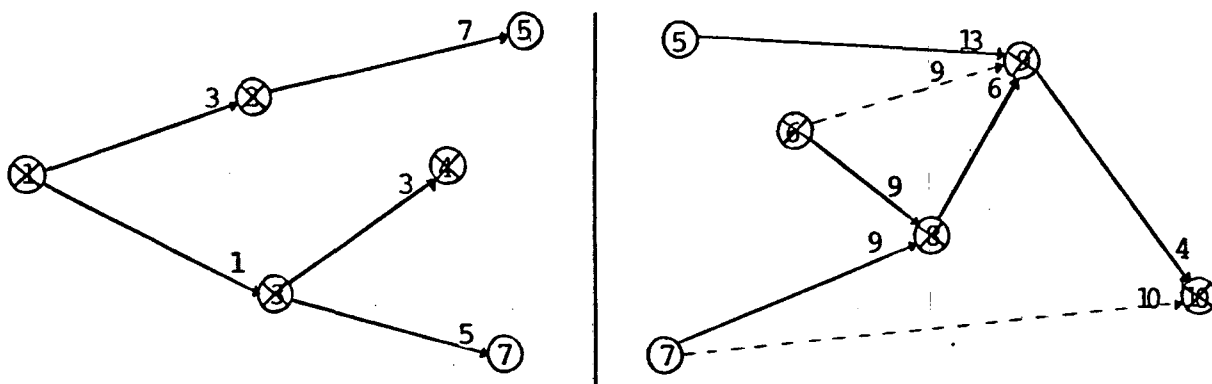


FIGURA 4.c

ITERAÇÕES 7 E 8:

Na iteração 7, os resultados são os seguintes:

$$G(7) = 4, PAI(7) = 4 \text{ e}$$

$$G(5) = 7; PAI(5) = 2 \text{ da iteração 5}$$

O custo associado ao caminho até o vértice 7 foi alterado, pois o caminho através dos vértices 4 é melhor que o anterior.

então $G(K) = \min [7, 4]$

$$K = 7$$

$$MIN = 15 > G(7)$$

portanto $k = 7$

$$\hat{G}(7) = 4$$

$$\text{FEXADO}(7) = 2$$

$$F = \{1, 10, 3, 9, 2, 8, 4, 6, 7\}$$

$$NF = 9$$

como $\text{MIN} > G(7) + G'(7)$

$$15 > 4 + 9$$

então $\text{MIN} = 13$ e

$$M = k = 7$$

Como $NF \neq 10$, a pesquisa ainda prossegue.

Os resultados para a iteração 8 são os seguintes:

$$G'(5) = 11; \text{FILHO}(5) = 6$$

O caminho para o vértice 5 teve seu custo alterado devido ao fato de que o seu custo associado é melhor através do vértice 6.

então $G'(U) = \min [11]$

$$U = 5$$

$$\text{MIN} = 13 > G'(5)$$

$$\text{MIN} = 13 > 11$$

portanto $u = 5$

$$G'(5) = 11$$

$$\text{FEXADO}(5) = 2$$

$$F = \{1, 10, 3, 9, 2, 8, 4, 6, 7, 5\}$$

$$NF = 10$$

Como $NF = 10$, o algoritmo se encerra e os resultados estão ilustrados na figura 4.d.

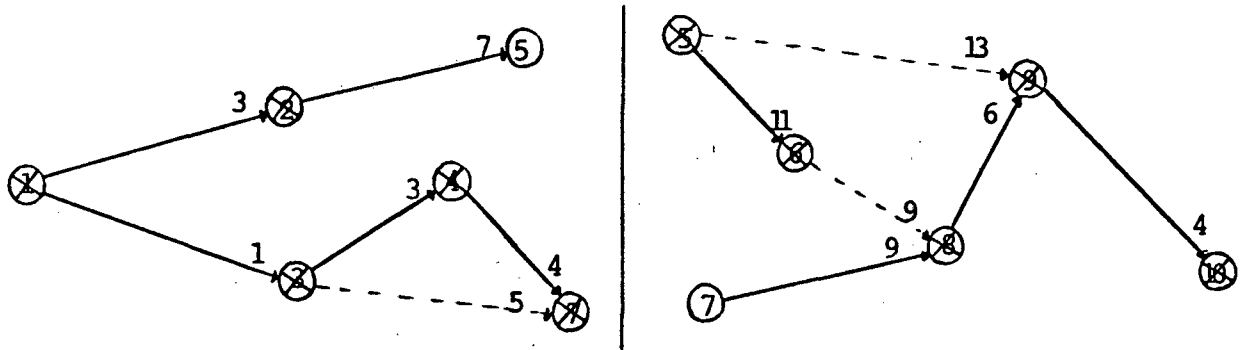


FIGURA 4.d

O caminho mínimo é formado a partir do vértice 7, pois no decorrer da pesquisa, $M = 7$. Para a primeira parte do caminho, este vértice é considerado como sendo o último, que através do vetor indicador PAI, obtém-se o seguinte:

$$7, 4, 3, 1;$$

e para a segunda parte, o vértice 7 é o nó inicial, que através do vetor indicador FILHO, obtém-se o seguinte:

$$7, 8, 9, 10;$$

que anexado a primeira parte devidamente ordenado, o caminho fica completo, assim:

$$\gamma = (1, 3, 4, 7, 8, 9, 10)$$

e o seu custo associado é:

$$C = 13$$

O resultado final está ilustrado na figura 4.e.

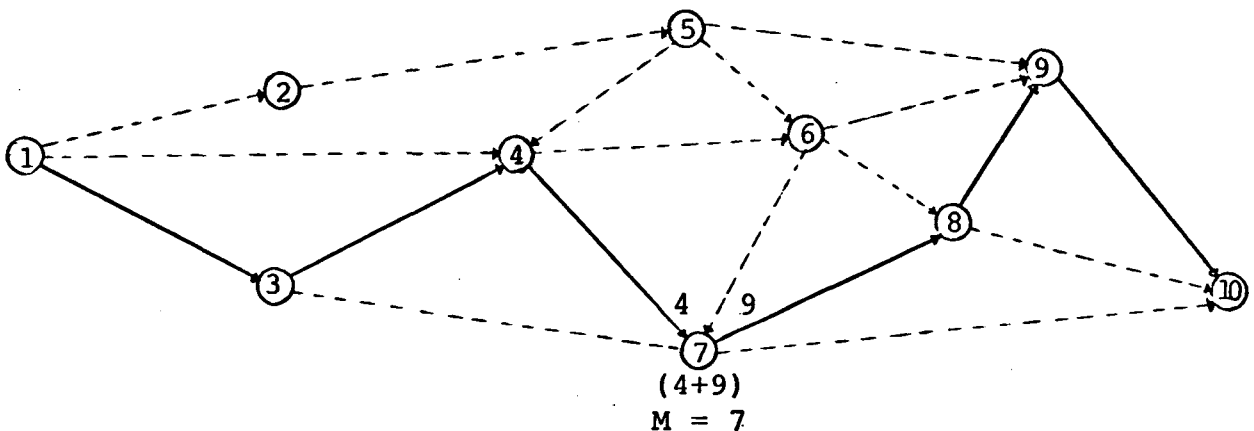


FIGURA 4.e

COMENTÁRIOS:

- este algoritmo realiza a pesquisa dos vértices nos dois sentidos: tanto do início ao final, como do final ao início, rotulando-os como temporários ou permanentes;

- portanto, se faz necessário a utilização de dois sistemas de apontadores, bem como a verificação da existência de um ponto em comum entre os mesmos;

- conseqüentemente, a reconstituição do caminho também é efetuada em duas etapas: a primeira a partir do ponto comum até o vértice inicial $s \in S$; a segunda, a partir do ponto comum até o vértice terminal $t \in T$.

C A P Í T U L O I V

4. ANÁLISE DOS RESULTADOS E CONCLUSÕES

4.1. DESCRIÇÃO DOS PROBLEMAS RESOLVIDOS:

Para realizar uma análise comparativa quanto ao desempenho computacional entre os algoritmos de Dijkstra e Jeyaratnam, estes foram implementados na linguagem de programação WATFIV e testados com grafos dos mais diferentes números de vértices (figuras de 6 a 16 - Anexo 1).

Os problemas analisados neste trabalho foram gerados aleatoriamente, sem a preocupação em estabelecer regras quanto a numeração dos vértices, quanto ao número de sucessores ou quanto a alocação dos custos aos arcos.

Na elaboração dos problemas, manteve-se sempre a preocupação básica de não beneficiar através deles nenhum dos algoritmos pesquisados, evitando-se a ocorrência de tendenciosidade. Isso foi obtido através de geração de problemas com diversos números de nós, diferentes graus de conexidade e sempre com grafos orientados anti-simétricos.

4.2. ANÁLISE DOS RESULTADOS OBTIDOS

Uma análise comparativa dos resultados obtidos quanto ao desempenho computacional dos algoritmos de Dijkstra e Jeyaratnam é apresentado neste capítulo.

Para tal, inicialmente, os algoritmos foram implementados em linguagem FORTRAN IV e testados num computador IBM 4341, utilizando o próprio compilador FORTRAN. Posteriormente os algoritmos foram testados no compilador WATFIV, para aproveitar as facilidades de obtenção dos tempos de compilação e de execução.

Ambos os programas foram submetidos a testes com 20 redes de 10,15,20,25,30,35,40,45 e 50 vértices, e ao final, com mais 20 redes de 100 e 200 vértices.

Primeiro, cada problema foi testado individualmente com os dois algoritmos. Em seguida, os testes foram realizados com série de 20 problemas e considerando diferentes números de vértices. Ao final calculou-se a média aritmética dos tempos de execução para cada grupo.

Como não houve diferença entre os resultados obtidos com os testes realizados para cada problema e os realizados em grupo, para este trabalho, optou-se para análise, pelos resultados obtidos em série, embora ambos os resultados estejam sendo analisados.

Os resultados utilizados para a análise foram obtidos com ambos os algoritmos sendo submetidos a testes nas mais variadas situações de carga de Unidade Central do Computador: livre, sobrecarregado e semi-carregado.

Para os problemas individuais, os resultados obtidos com o algoritmos de Dijkstra apresentou uma ligeira vantagem sobre os de Jeyaratnam, para grafos com poucos vértices e, à medida que aumentava o número de vértices, a diferença também aumentava.

Nos testes em série, mesmo sendo dados apenas um tempo de execução, e fornecendo para cada problema o caminho de mínimo custo, também neste caso, os resultados demonstraram a maior eficiência do algoritmo de Dijkstra.

A tabela 1 apresenta os resultados para testes realizados individualmente, cada problema foi testado três vezes e ao final calculou-se os tempos médio de execução.

A tabela 2 apresenta os resultados obtidos para os problemas testados em série de 20 grafos, que ao final, também foi calculado os respectivos tempos médio de execução e diferença.

NÚMERO DE NÓS	TEMPO MÉDIO EXECUÇÃO		DIFERENÇAS - (D-J)	
	DIJKSTRA	JEYARATNAM	ABSOLUTA	%
10	0.160	0.162	- 0.002	1,25
15	0.234	0.240	- 0.006	2,56
20	0.326	0.333	- 0.013	3,99
25	0.429	0.456	- 0.027	6,29
30	0.554	0.589	- 0.035	6,32
35	0.711	0.750	- 0.039	5,48
40	0.909	1.000	- 0.091	10,01
45	1.084	1.220	- 0.136	12,54
50	1.274	1.482	- 0.208	16,33
100	4.612	4.956	- 0.344	7,46
200	17.531	19.001	- 1.470	8,38

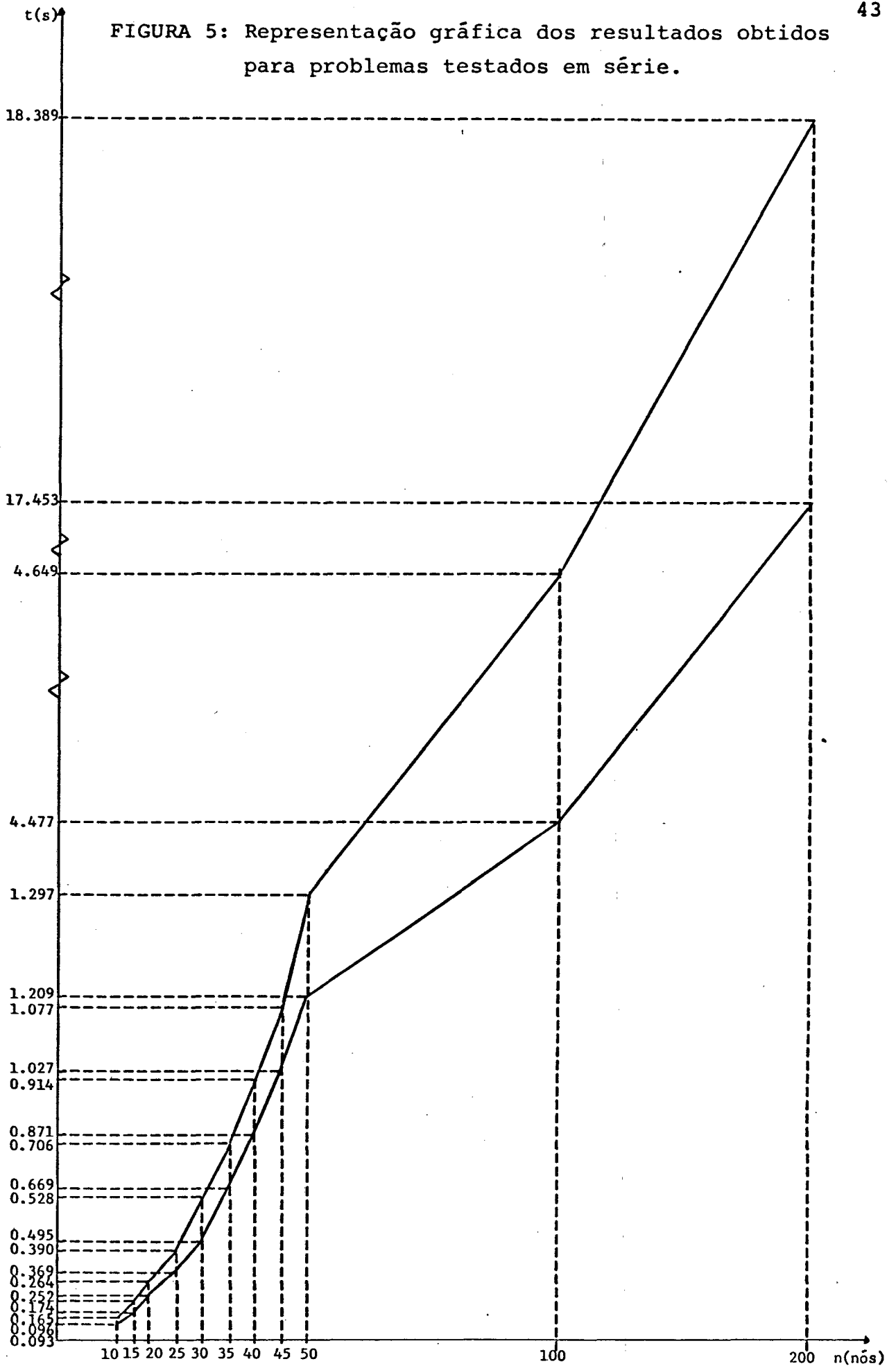
TABELA 1. Resultados para testes realizados com problemas individuais, com o tempo dado em segundos.

Nº PROB.	Nº NÓS	TEMPO DE EXECUÇÃO		TEMPO MÉDIO		DIFERENÇAS -(D-J)	
		DIJKSTRA	JEYARA- TNAM	DIJKSTRA	JEYARA- TNAM	ABSOLUTA	%
20	10	1.87	1.92	0.093	0.096	-0.003	3,23
20	15	3.30	3.42	0.165	0.174	-0.009	5,45
20	20	5.04	5.27	0.252	0.264	-0.012	4,76
20	25	7.38	7.79	0.369	0.390	-0.021	5,69
20	30	9.90	10.55	0.495	0.528	-0.033	6,67
20	35	13.38	14.12	0.669	0.706	-0.037	5,53
20	40	17.41	18.27	0.871	0.914	-0.043	4,94
20	45	20.54	21.54	1.027	1.077	-0.050	4,87
20	50	24.17	25.94	1.209	1.297	-0.088	7,28
20	100	89.54	93.88	4.477	4.649	-0.217	4,85
20	200	349.05	367.78	17.453	18.389	-0.936	5,36

TABELA 2. Resultados obtidos para problemas testados em série, com o tempo dado em segundos.

Observa-se na tabela que os resultados obtidos pelo algoritmo de Dijkstra são aparentemente melhores que os obtidos pelo Jeyaratnam. Nota-se também que a medida em que aumenta o número de vértices, a diferença tende a aumentar e não se pode dizer que o algoritmo de Jeyaratnam seja competitivo com o de Dijkstra para grafos com custos não-negativos.

FIGURA 5: Representação gráfica dos resultados obtidos para problemas testados em série.



4.3. CONCLUSÕES

Ao realizar um estudo preliminar do artigo de Jayaratnam, tudo levava a crer que o seu algoritmo poderia ser competitivo com o de Dijkstra, considerado como o mais eficiente para determinar um caminho de mínimo custo em grafos com custos não-negativos. O fato do algoritmo de Jayaratnam pesquisar em ambas as direções, tanto a partir de S como de T é que fornecia essa impressão.

Por esta razão, pensou-se que se esta hipótese se verificasse, em desenvolver um algoritmo associado com heurísticas⁹, como o algoritmo A*, gerado a partir do algoritmo de Dijkstra.

Com o desenvolvimento dos testes com o algoritmo de Jayaratnam, notou-se que ele não apresentou os resultados previstos, mesmo nas situações citadas pelo autor, em grafos com poucos vértices.

O fato do algoritmo de Jayaratnam rotular os vértices como permanentes em ambos os sentidos, tornou necessário desenvolver dois sistemas de apontadores e um ponto de encontro entre os sub-caminhos obtidos. Para isso, é preciso realizar comparações adicionais dos custos envolvidos sempre que se rotular um vértice como permanente, pois ele será o referencial na reconstituição do caminho de mínimo custo. Este vértice, entretanto, nem sempre se encontra no ponto intermediário do grafo e sendo assim, torna-se necessário efetuar mais operações e comparações do que aquelas desenvolvidas pelo algoritmo de Dijkstra.

Além disso, se o referido vértice não se encontrar em um dos vetores, significará que ele é o vértice inicial ou final, e conseqüentemente, um dos vetores armazenou vértices que não serão utilizados na reconstituição do caminho e portanto, "desnecessários".

Como o algoritmo de Jeyaratnam não apresentou os resultados esperados, a idéia inicial de desenvolver um algoritmo com heurísticas foi descartada.

E, como pode ser observado nas tabelas apresentadas, tanto para os problemas testados em separados como em conjunto, o algoritmo de Dijkstra mostrou ser realmente mais eficiente.

B I B L I O G R A F I A

1. ANDRADE, M.C.Q., A Criação no processo Decisório, Rio de Janeiro, 1980.
2. ARAÚJO, N.D., Análise de um Algoritmo de Fluxo em Redes com Remoção de Arcos, Tese de Mestrado - UFSC, 1981.
3. BERGE, C., Graphs and Hipergraphes, Amsterdã, 1970.
4. BOAVENTURA NETTO, P.O., Teoria e Modelos de Grafos, São Paulo, Edgard Blucher, 1979.
5. CHRISTOFIDES, N., Graph Theory, An Algorithmic Approach-Academic Press, 1975.
6. DANTZIG, G.B., Linear Programming and Extensions, New York, 1966.
7. DIJKSTRA, E.W., A Note on Two Problems in Connection With Graphs, Numer Math, 1 (1959) 269 - 271.
8. DREYFUS, S.E., An Appraisal of Some Shortest-Path Algorithms, Operational Reseach, 17 (1969) 395 - 412.
9. GONZAGA, C.C., Busca de Caminhos em Grafos e sua Aplicação a Problemas de Decisões Sequenciais. VIII Simpósio Brasileiro de Pesquisa Operacional, Recife, 1975.
10. JEYARATNAM, S., A New Algorithm for Finding the Shortest Path Between a Specified Pair of Nodes in a Graph of Nonnegative Arcs. Operational Research, 12 (1983) 375 - 778.
11. LUCCHERSI, C.L., Introdução à Teoria dos Grafos, 12ª Colô-

quio Brasileiro de Matemática, Poços de Caldas, 1979.

12. NICHOLSON, T.A.J., Finding the Shortest Rout Between Two Points in a Network, Computer Journal, 9 (1966) 275 - 280.

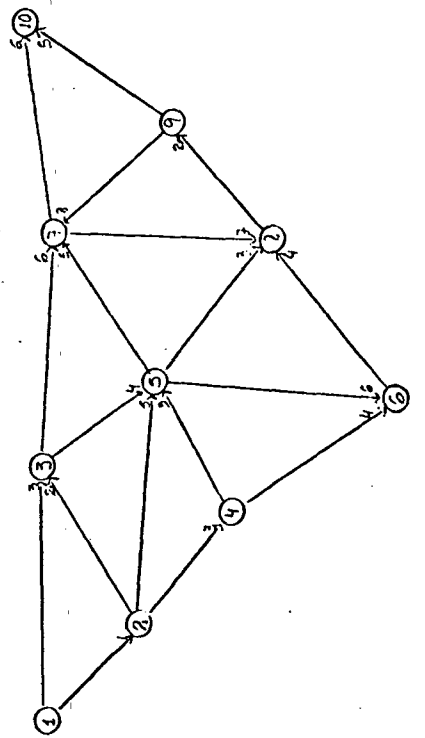
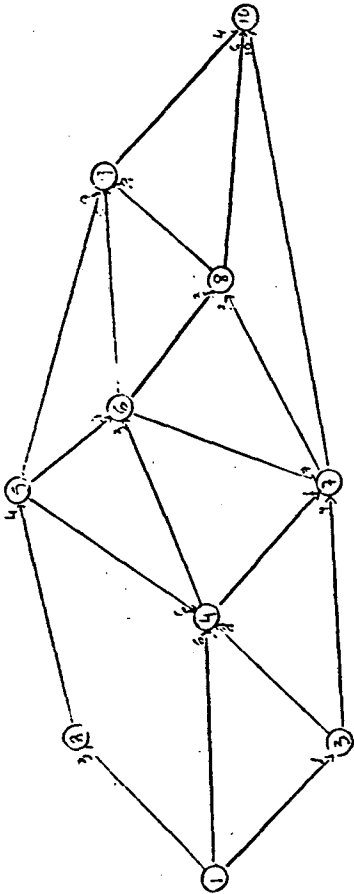
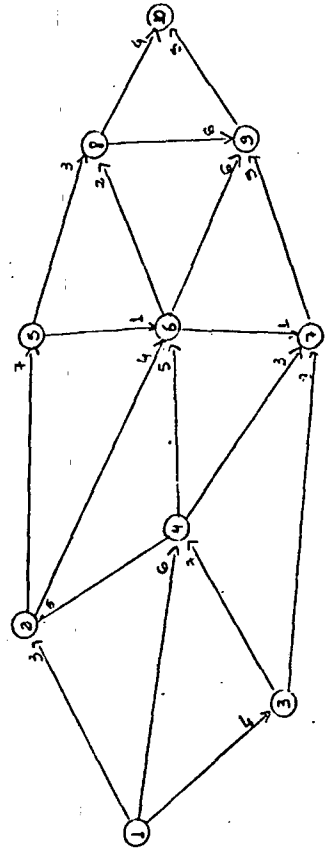
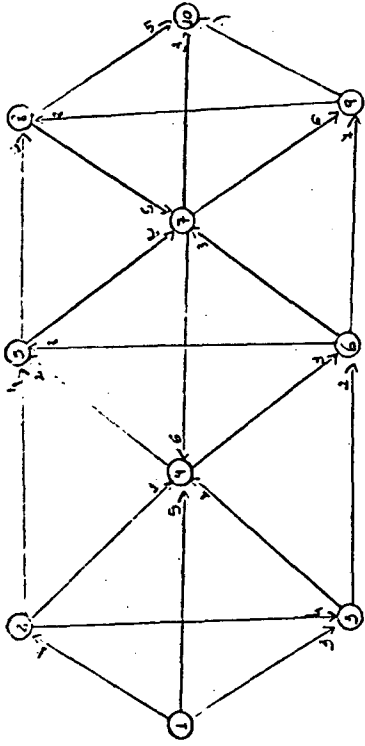
A N E X O S

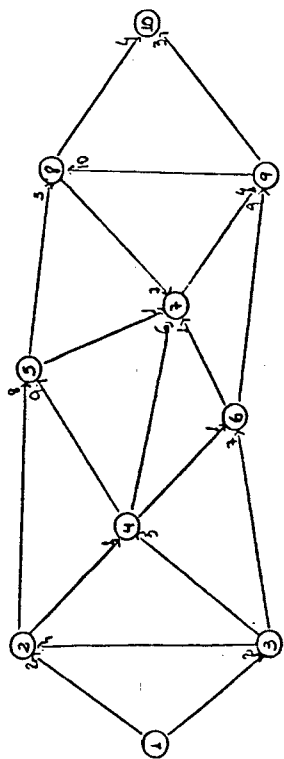
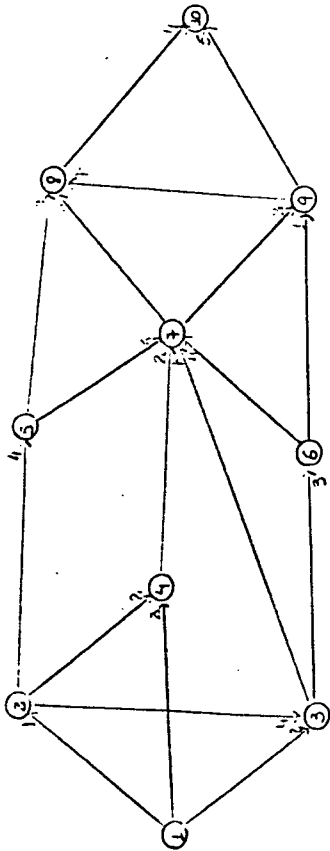
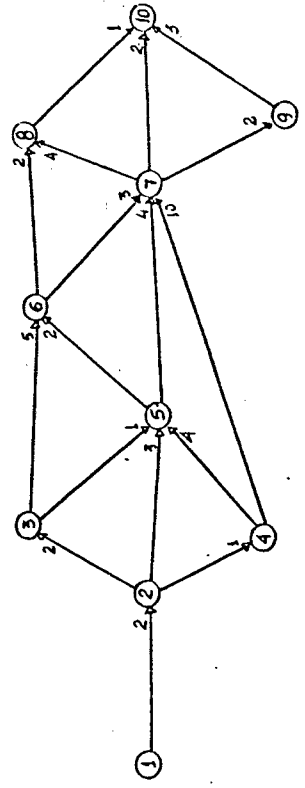
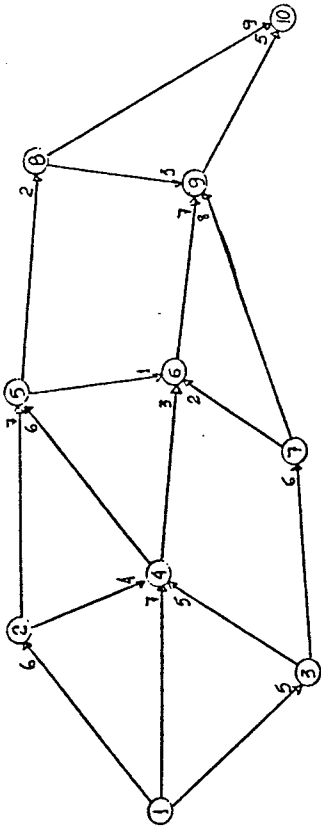
A N E X O 1

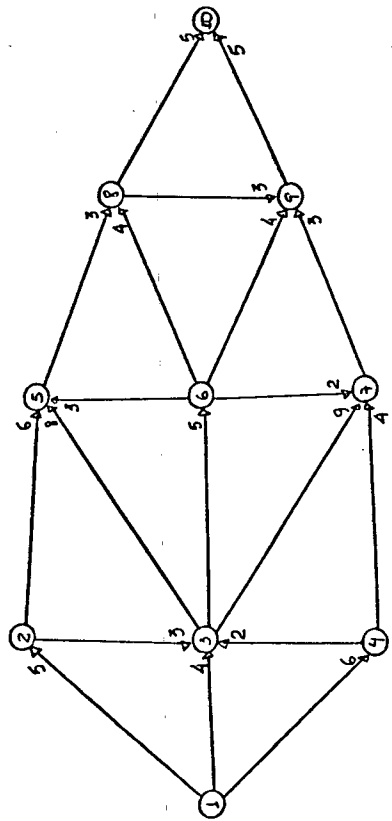
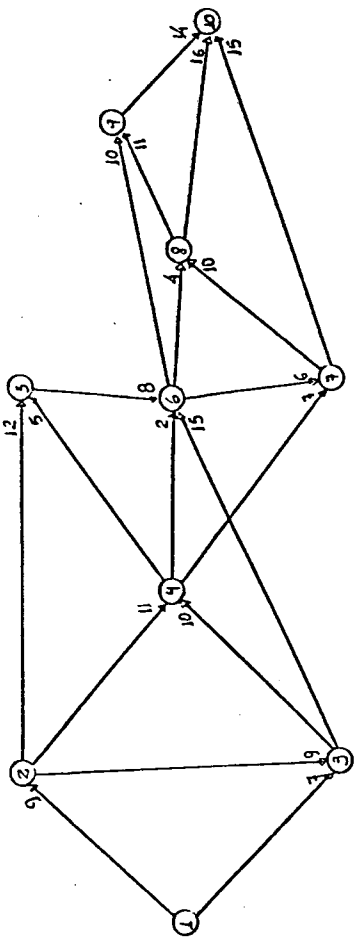
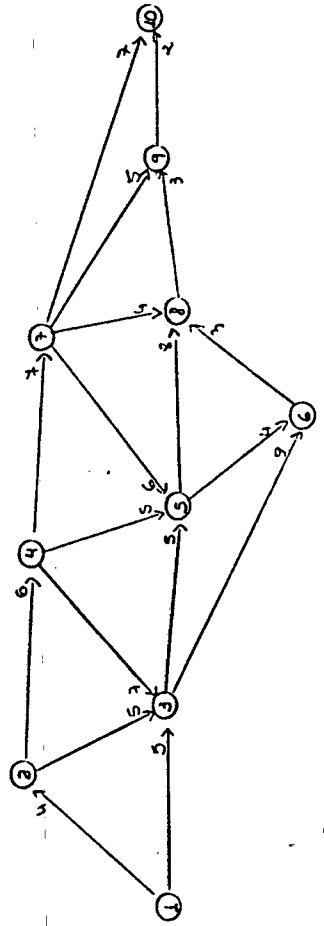
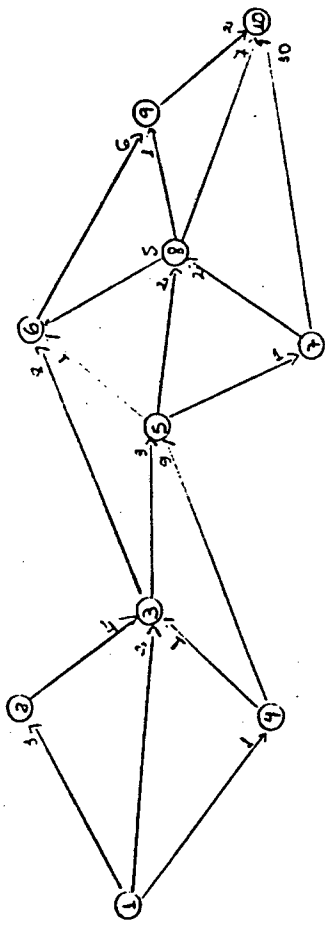
ILUSTRAÇÕES DOS PROBLEMAS RESOLVIDOS

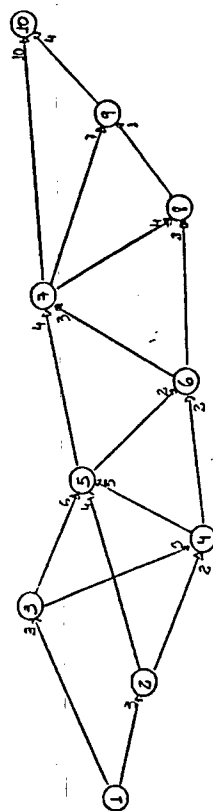
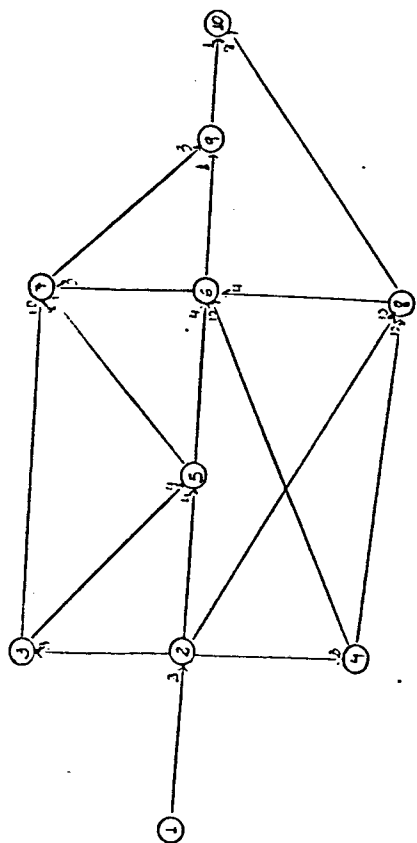
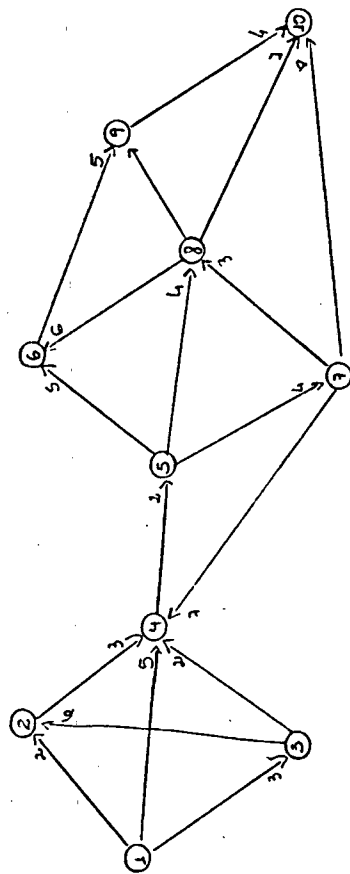
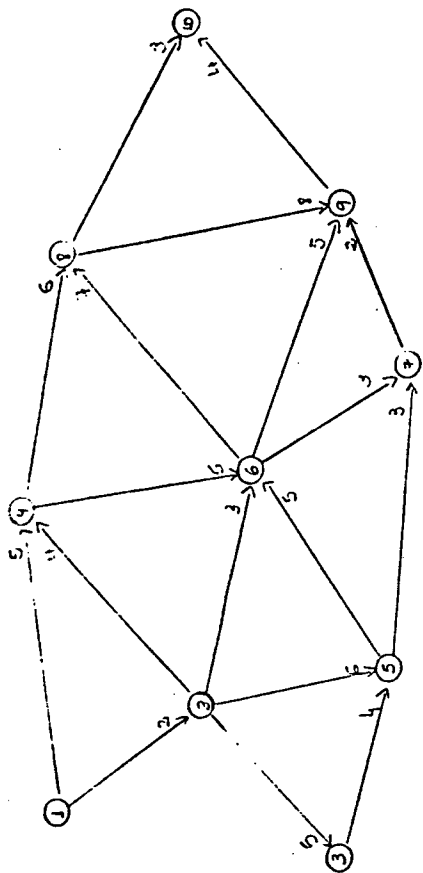
F I G U R A 6

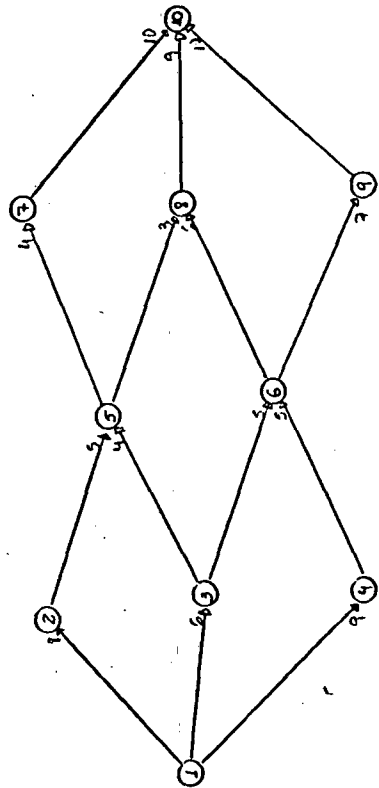
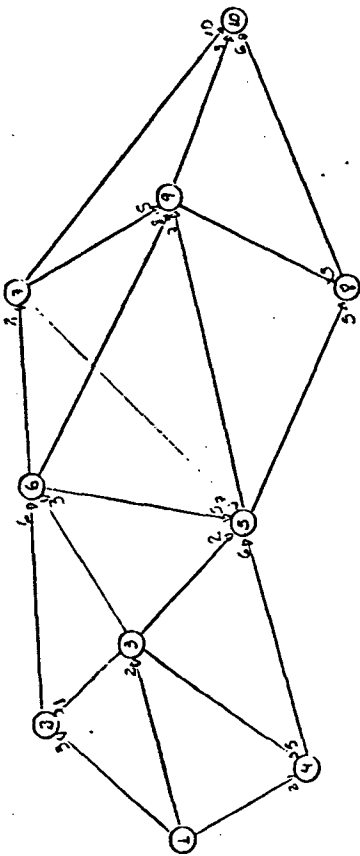
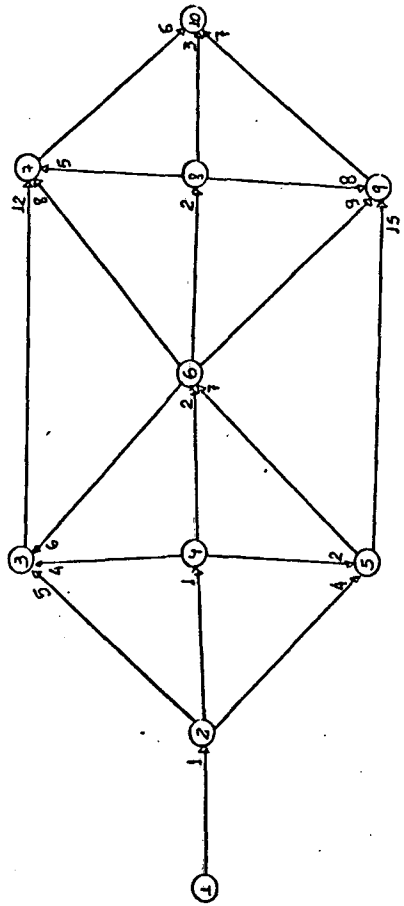
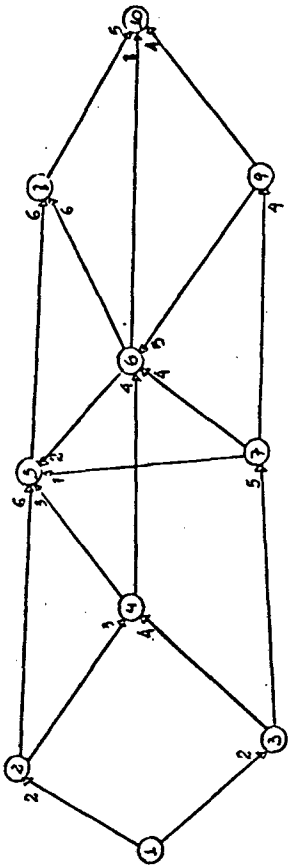
GRAFOS COM 10 VÉRTICES





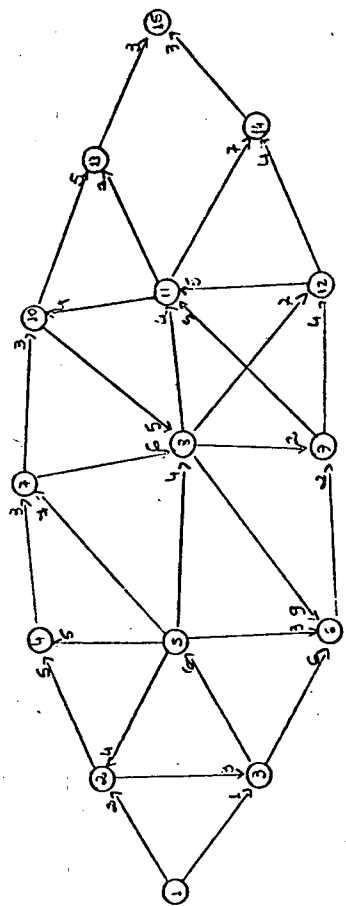
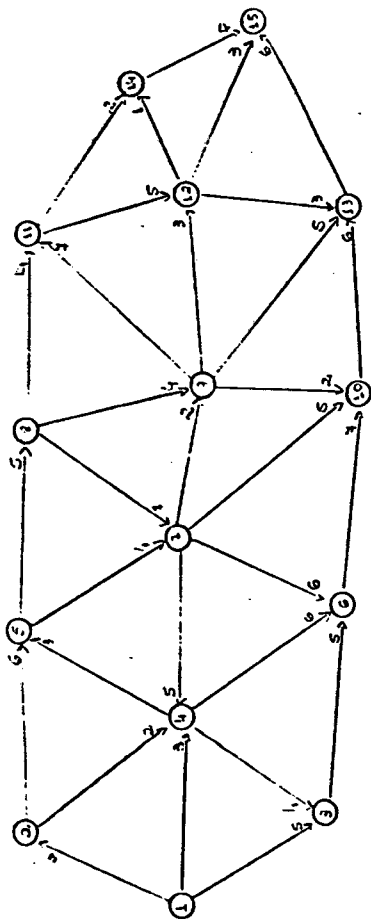
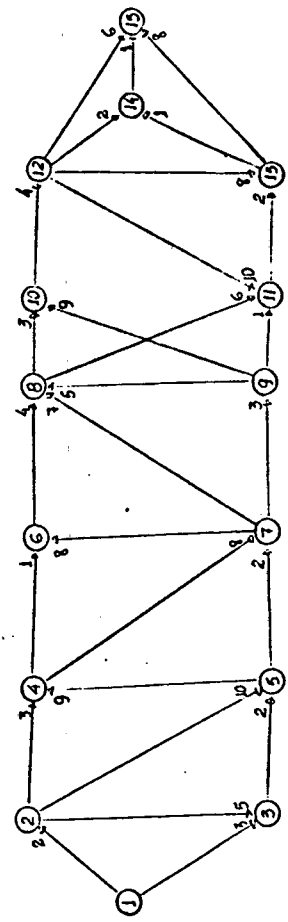
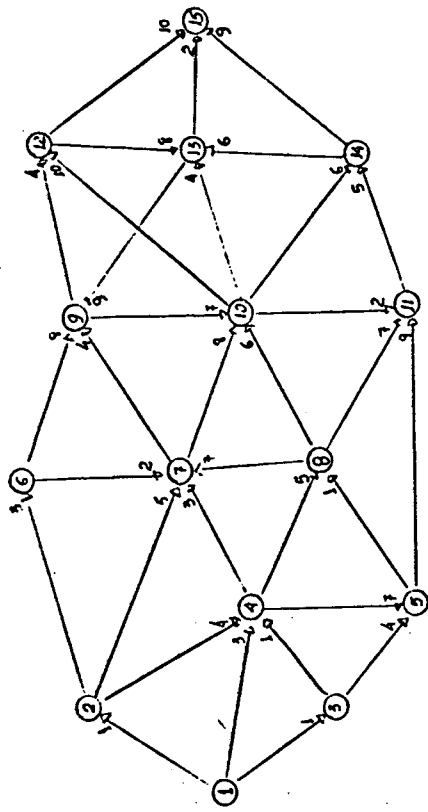


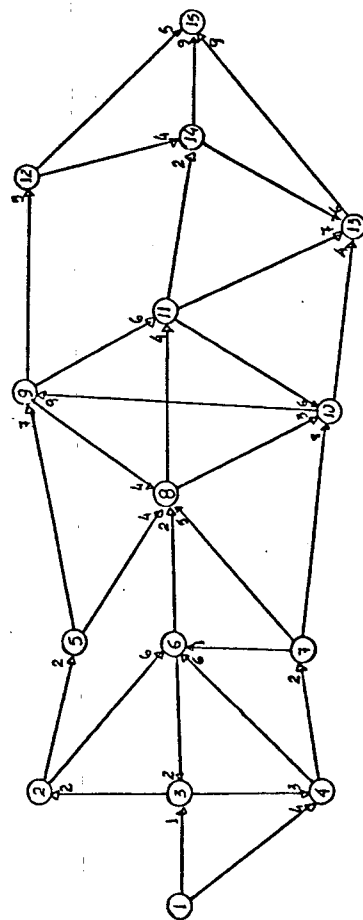
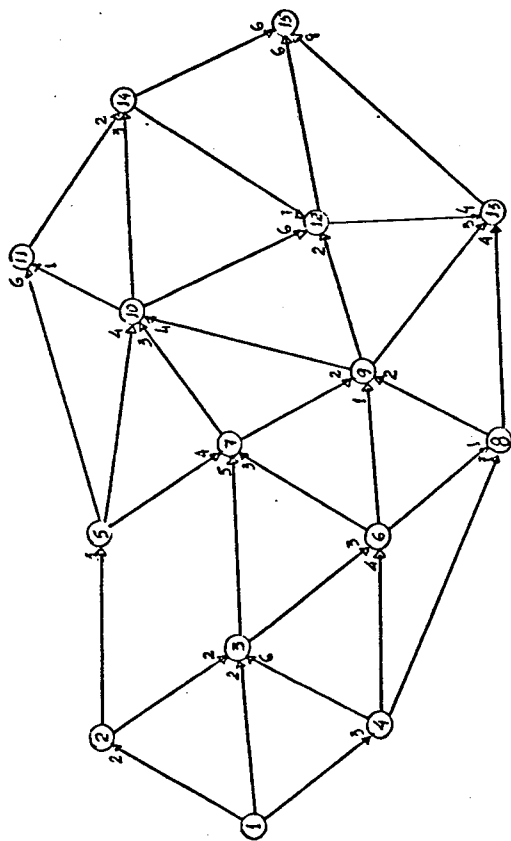
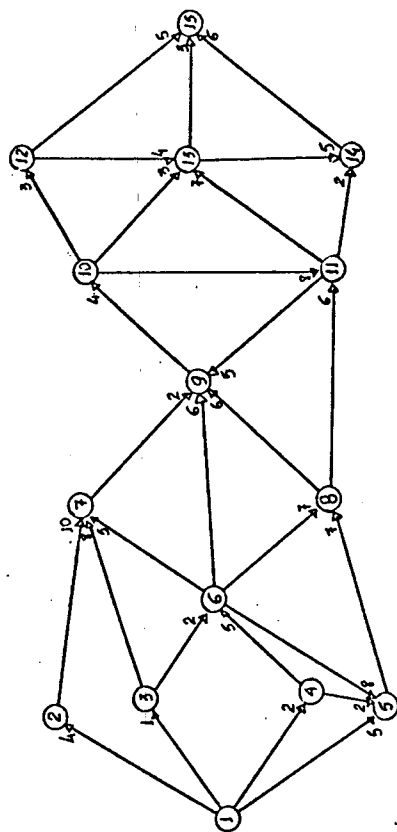
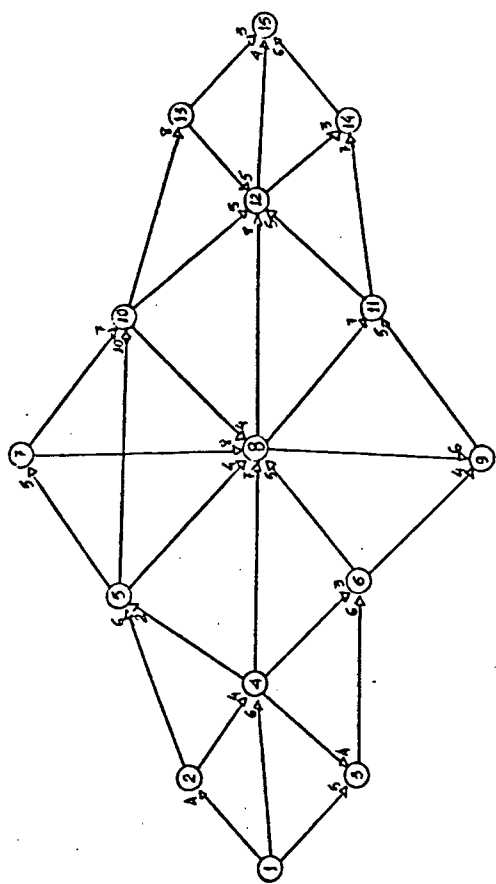


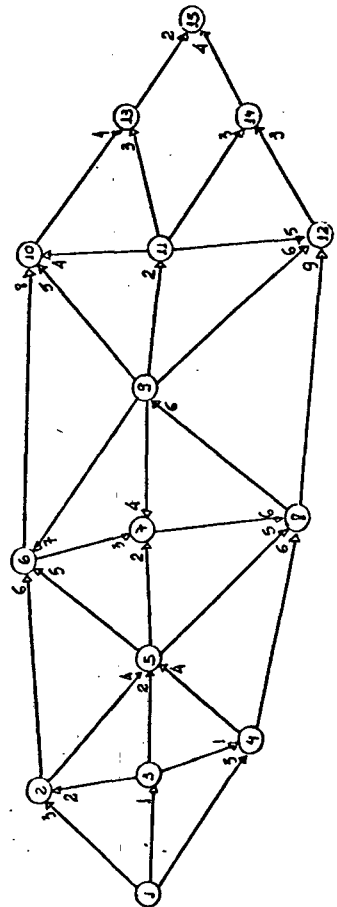
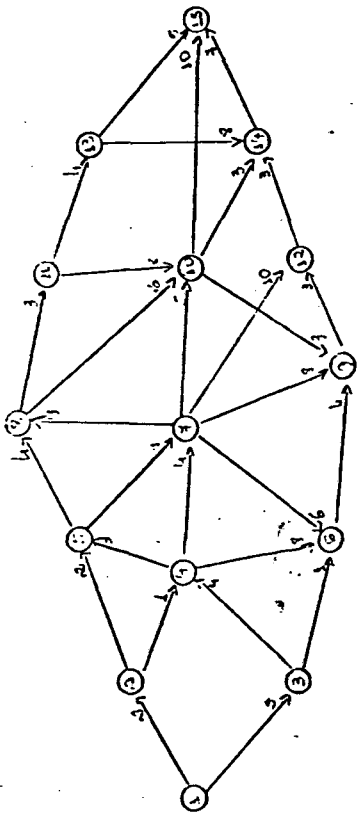
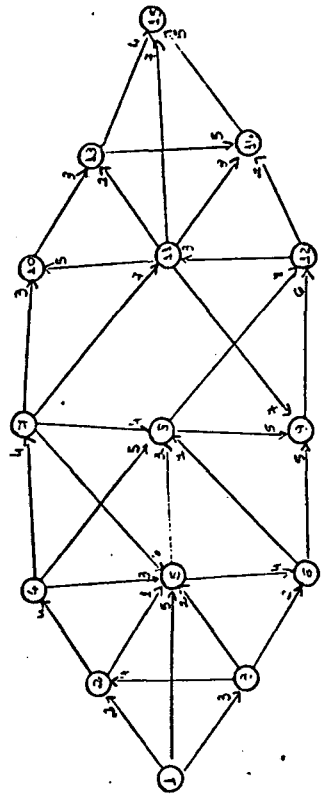
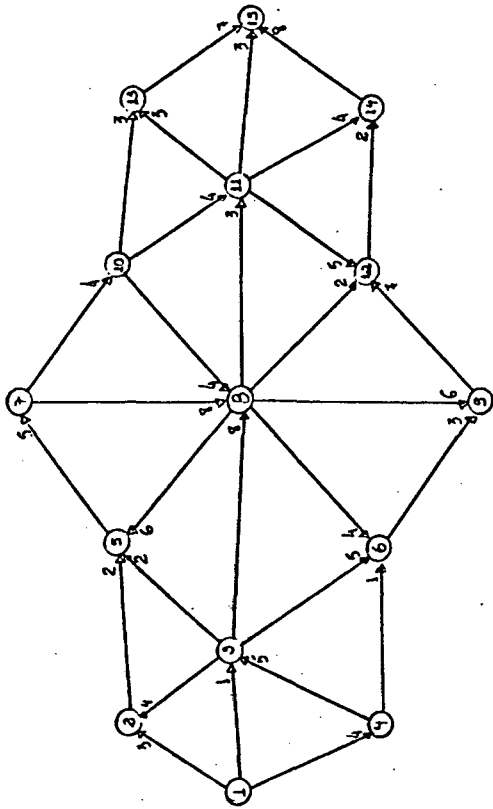


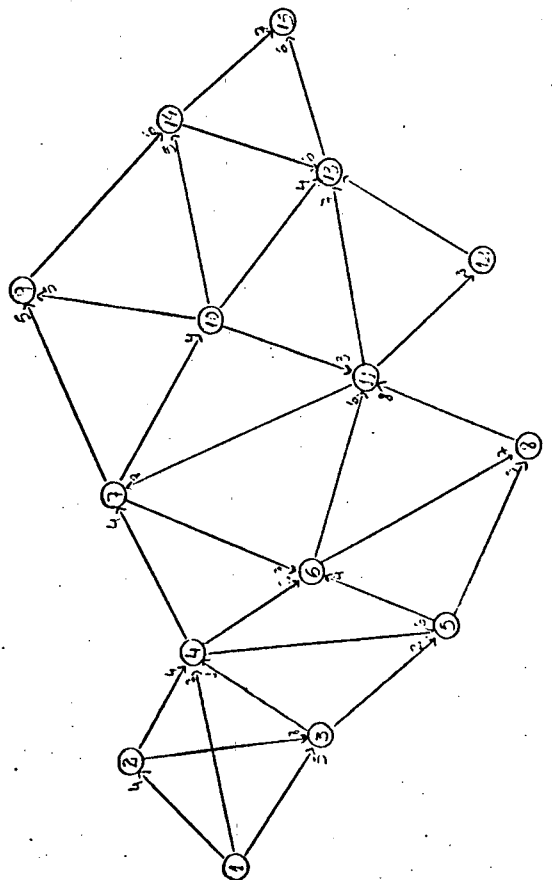
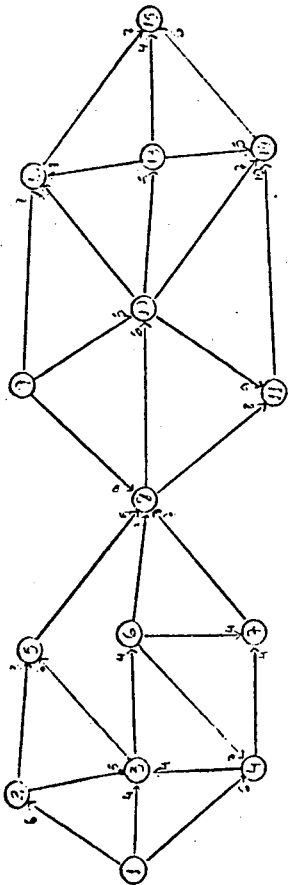
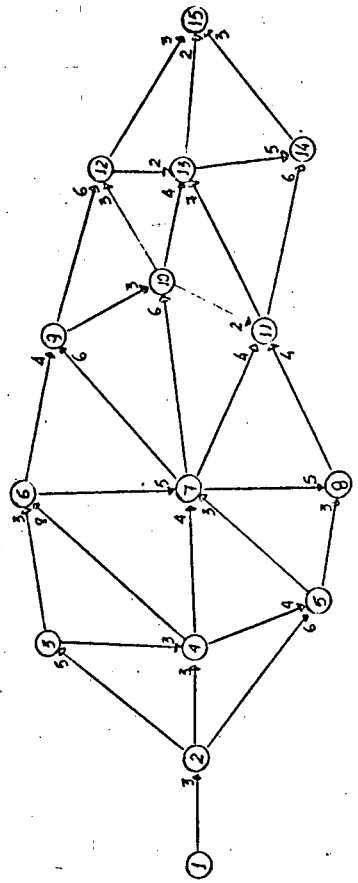
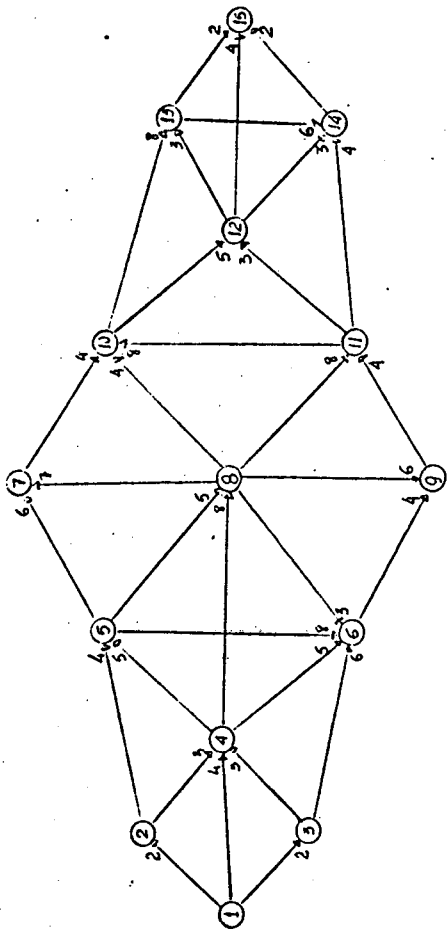
F I G U R A 7

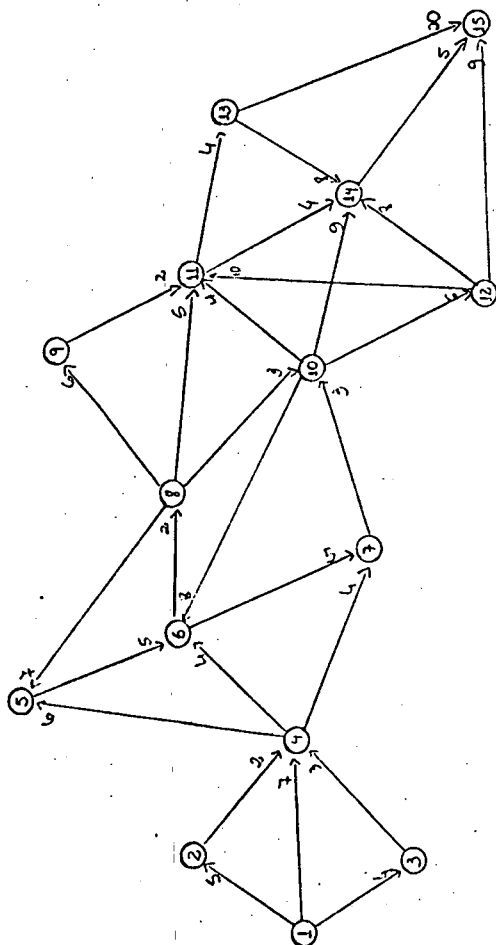
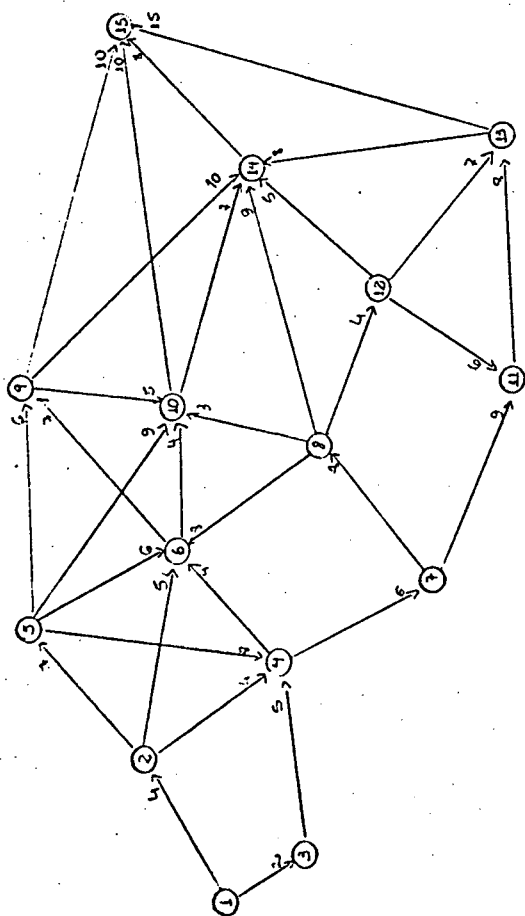
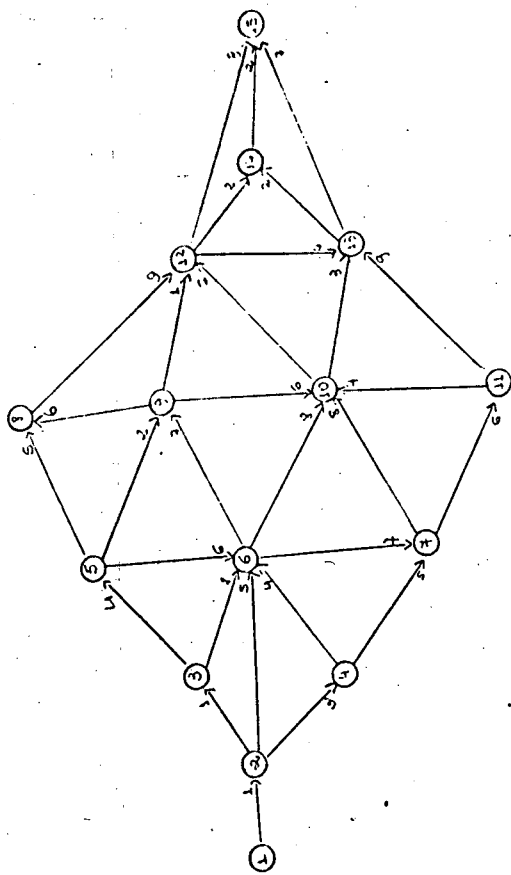
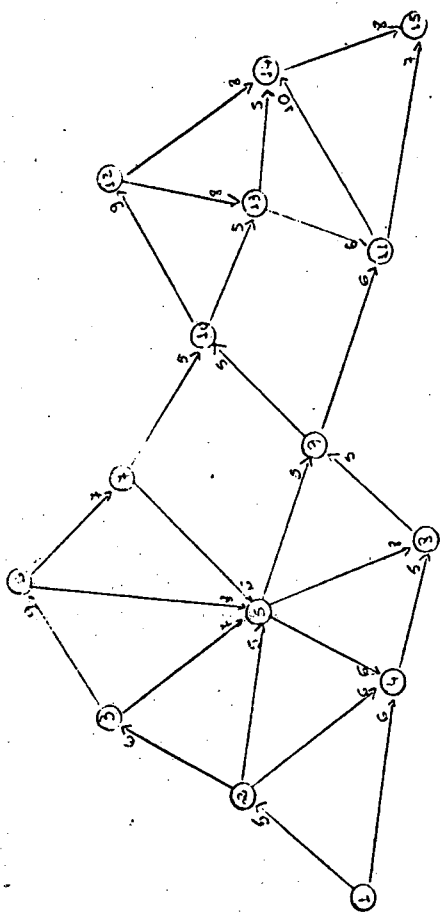
GRAFOS COM 15 VÉRTICES





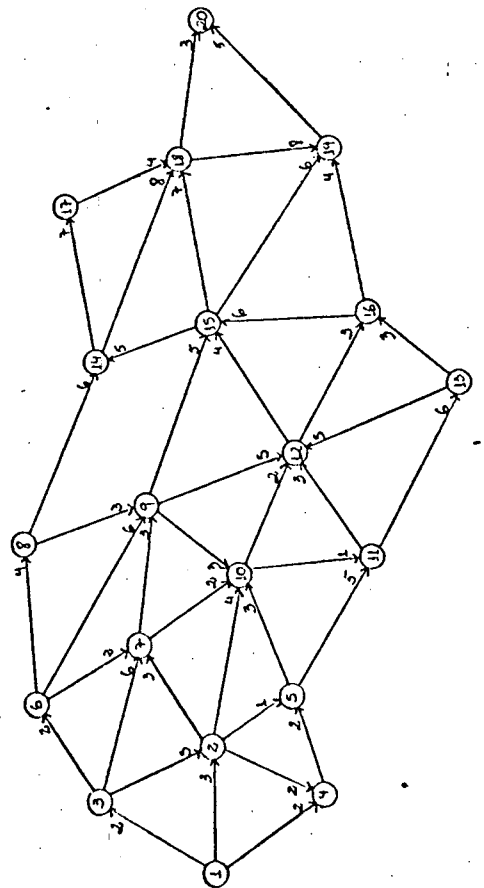
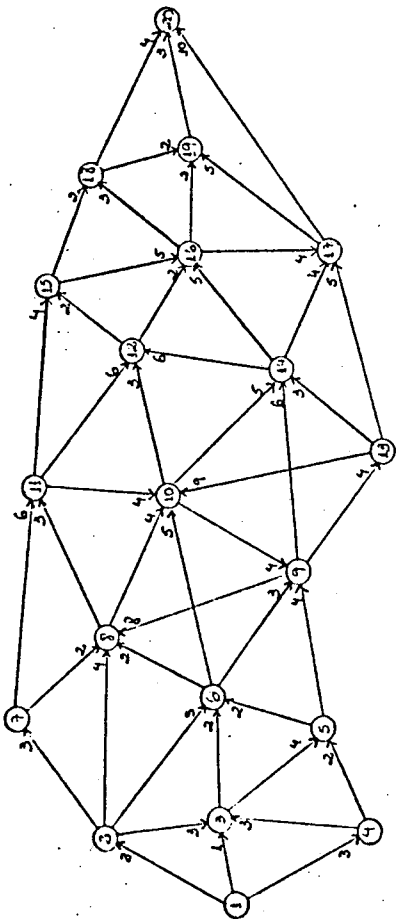
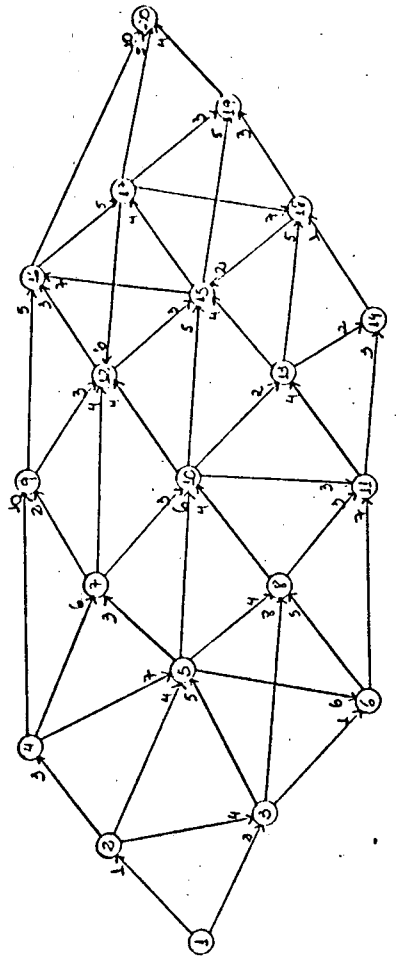
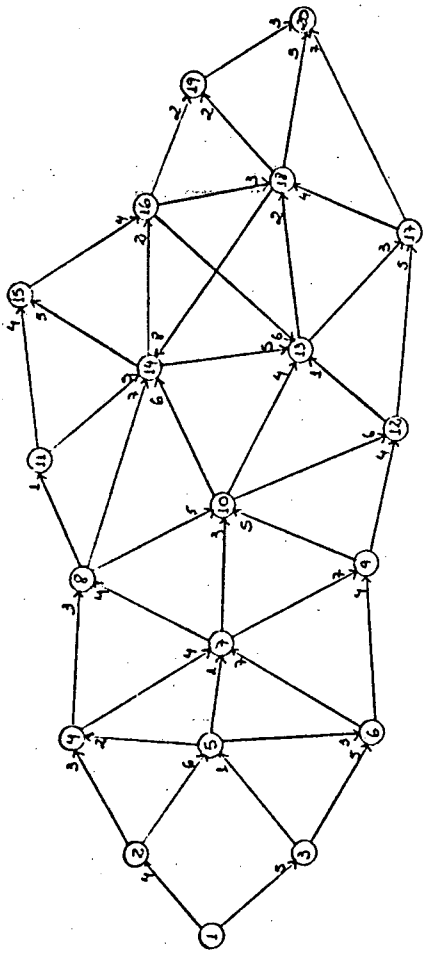


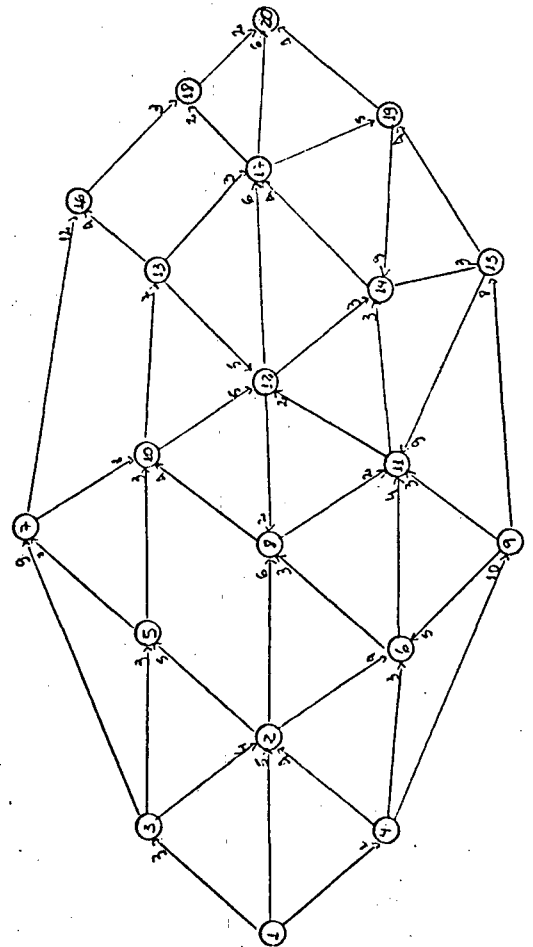
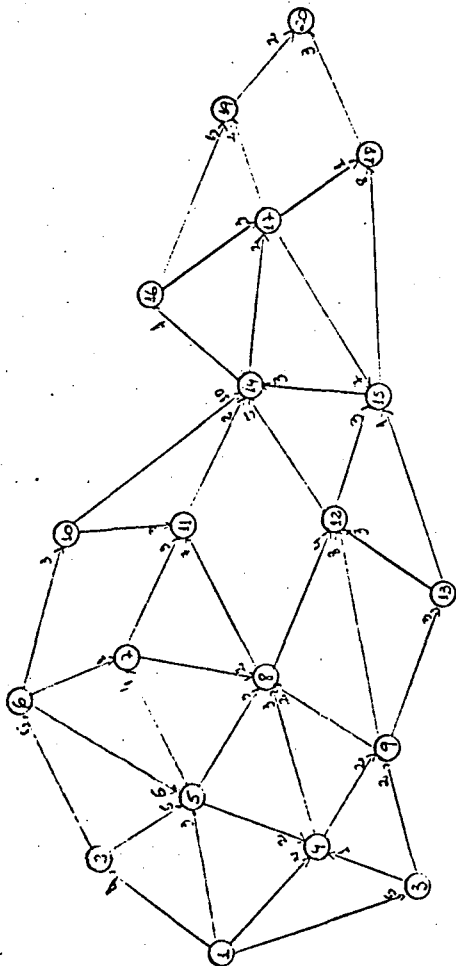
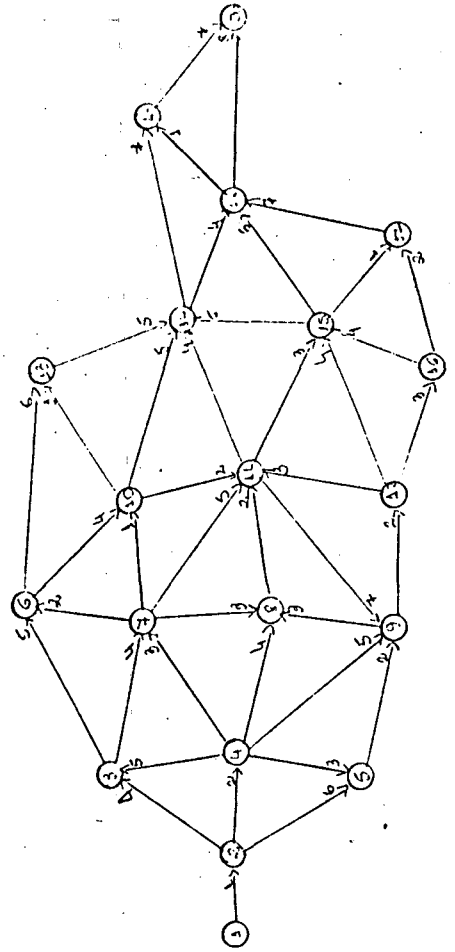
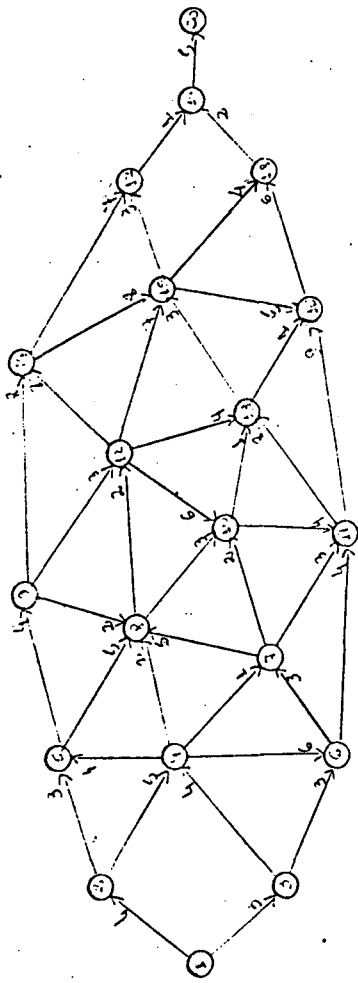


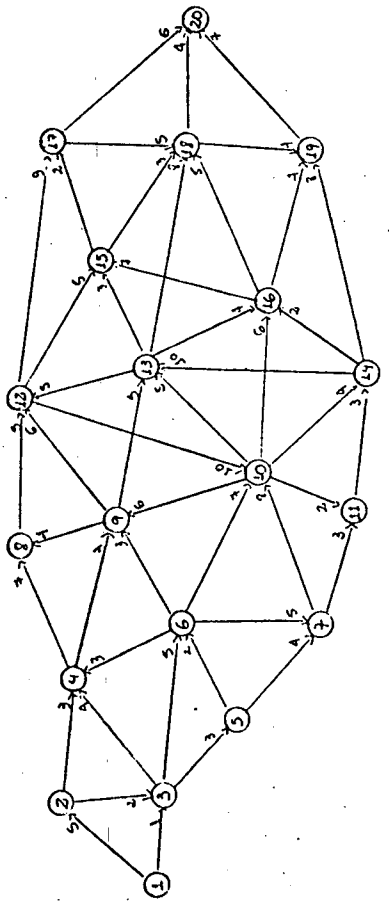
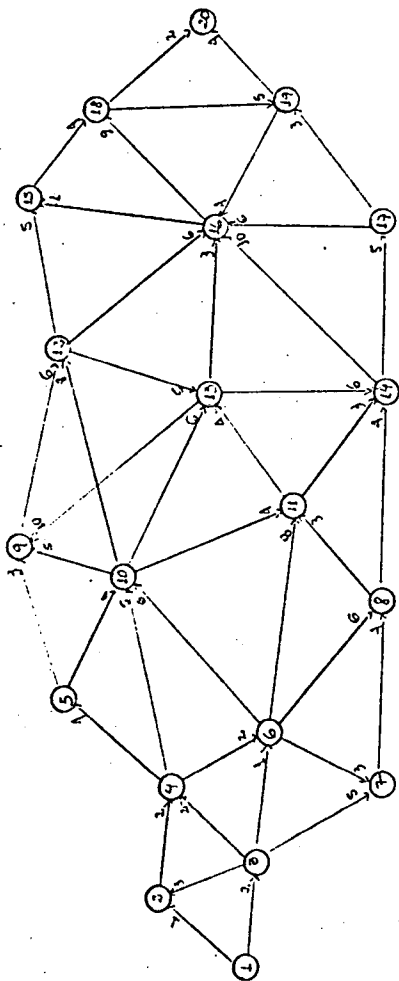
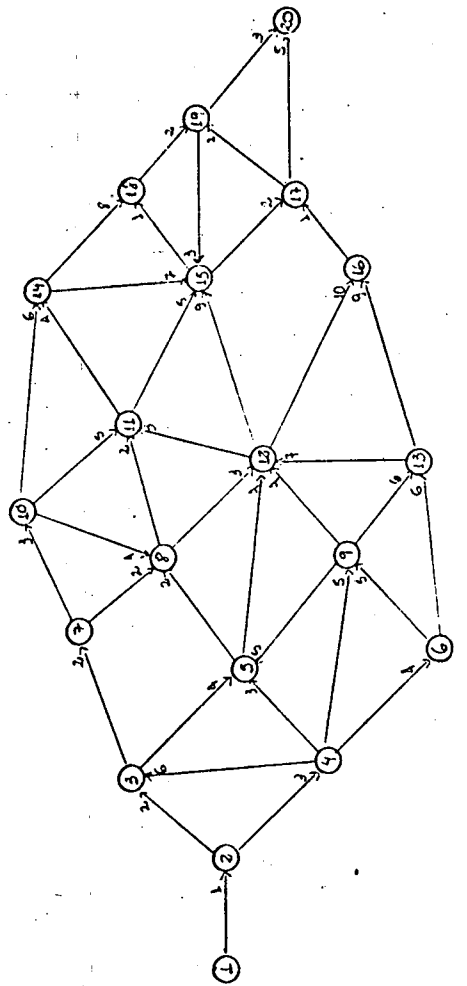
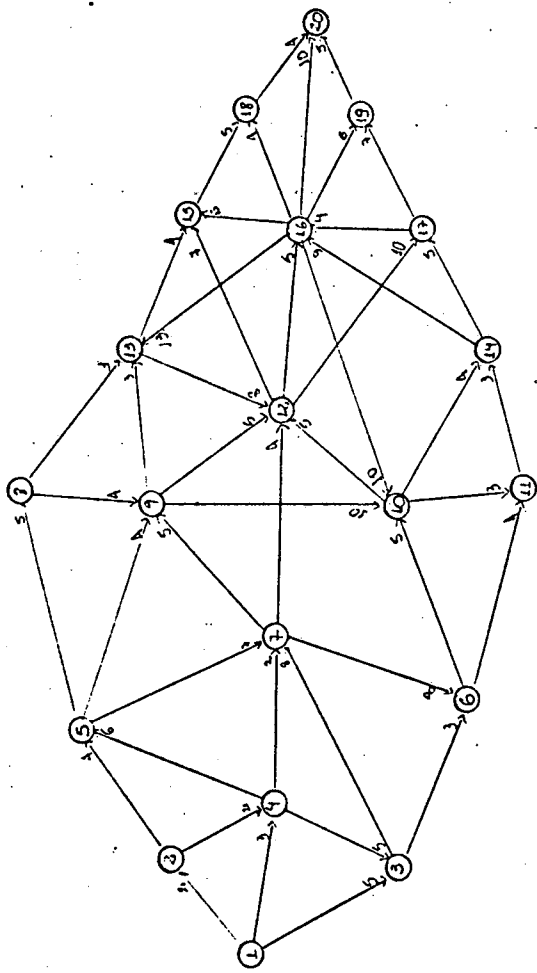


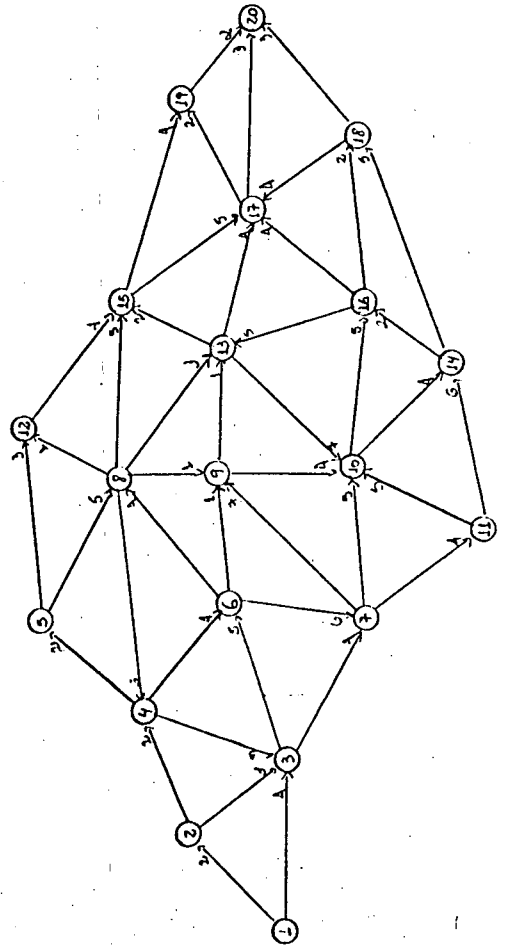
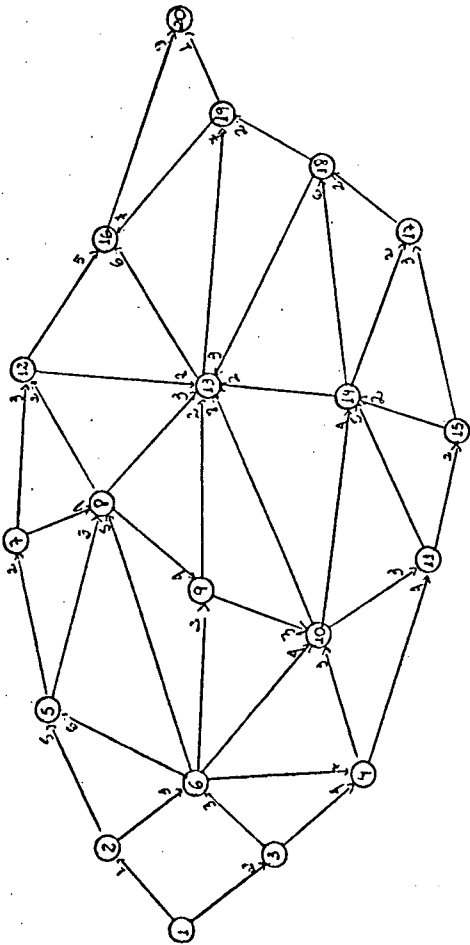
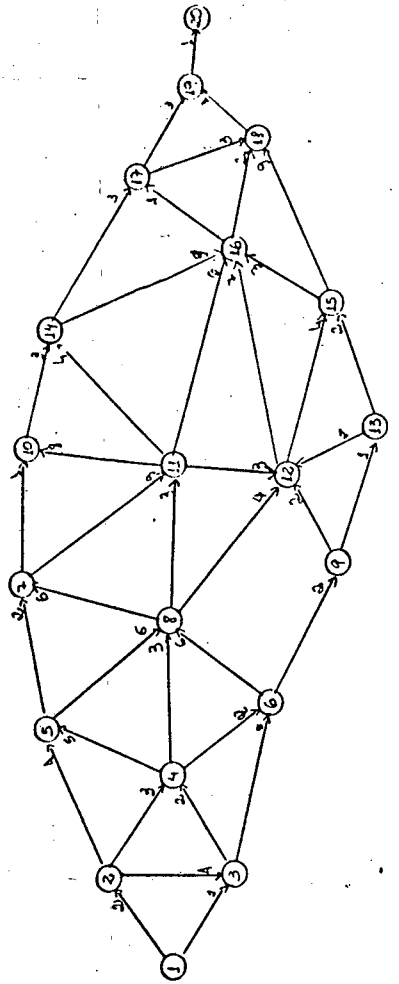
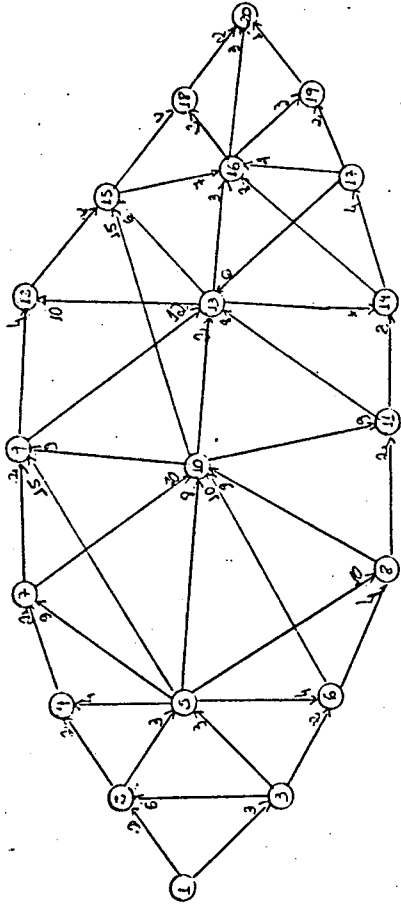
F I G U R A 8

GRAFOS COM 20 VÉRTICES









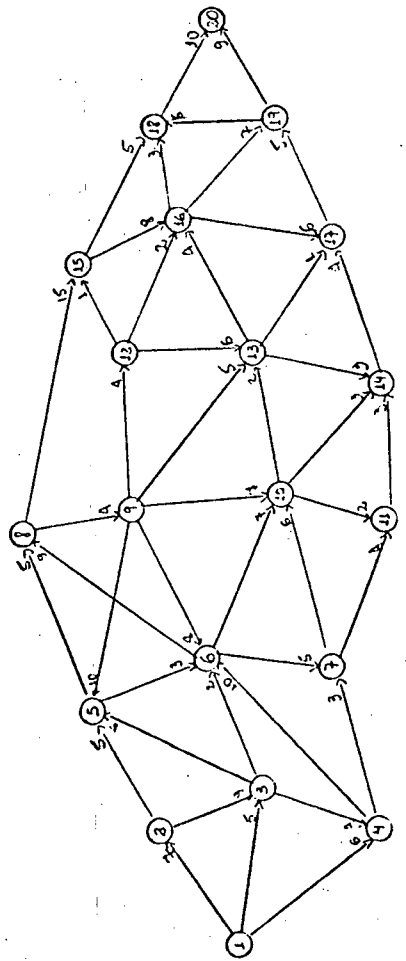
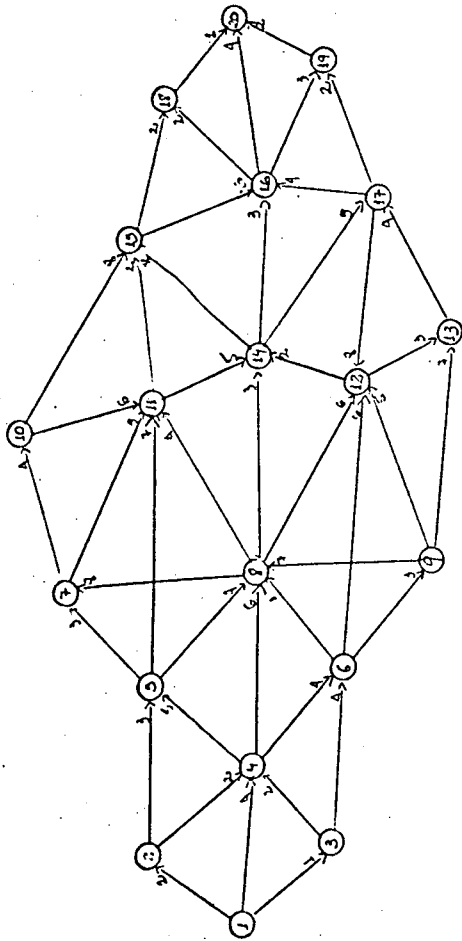
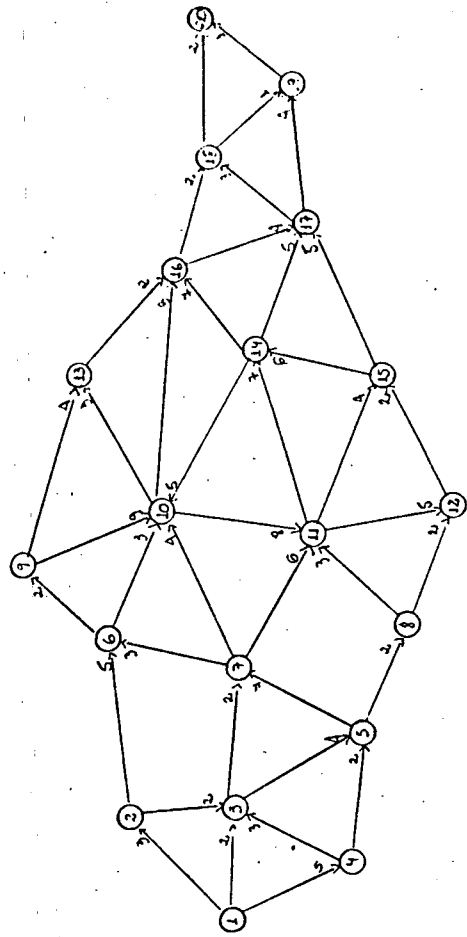
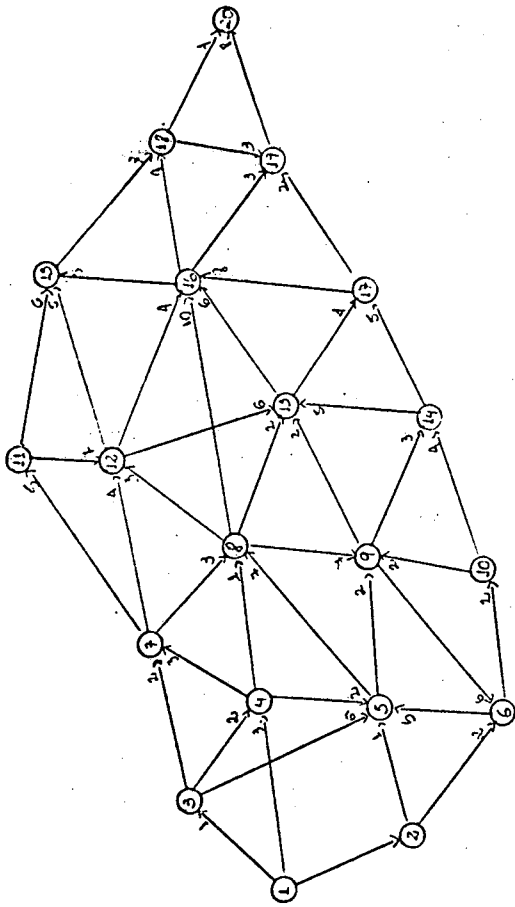
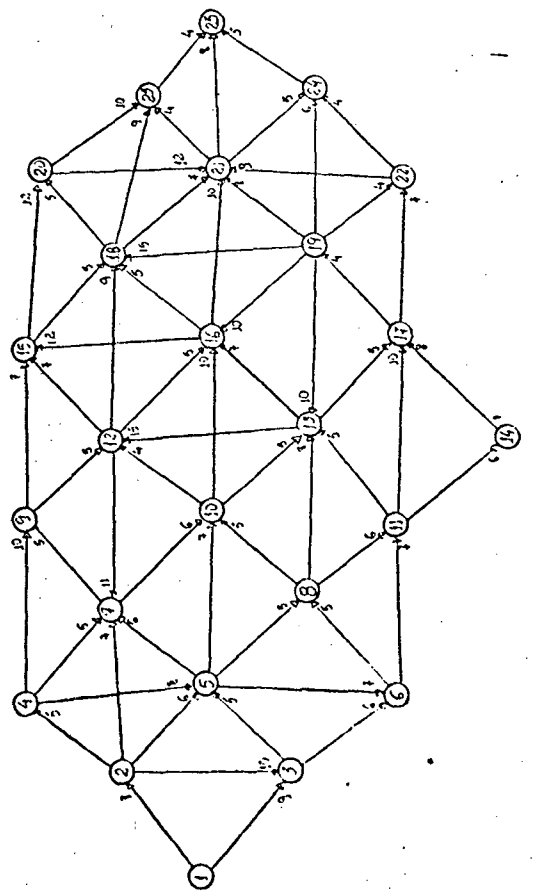
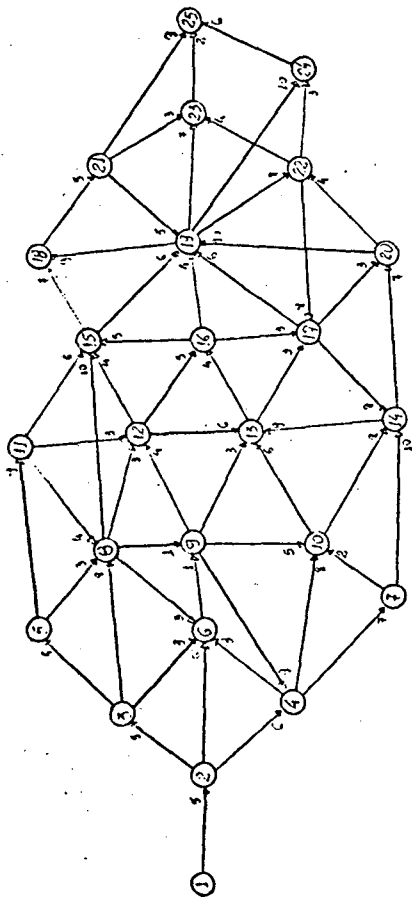
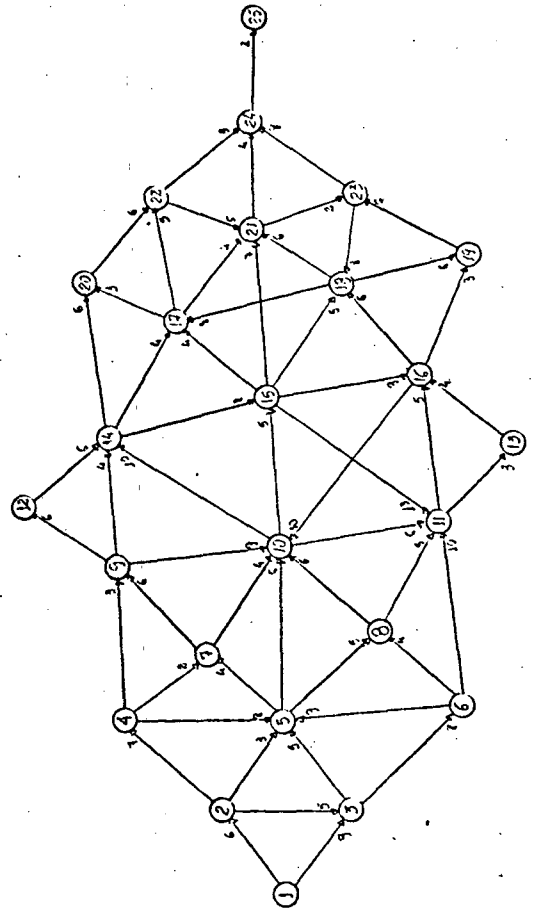
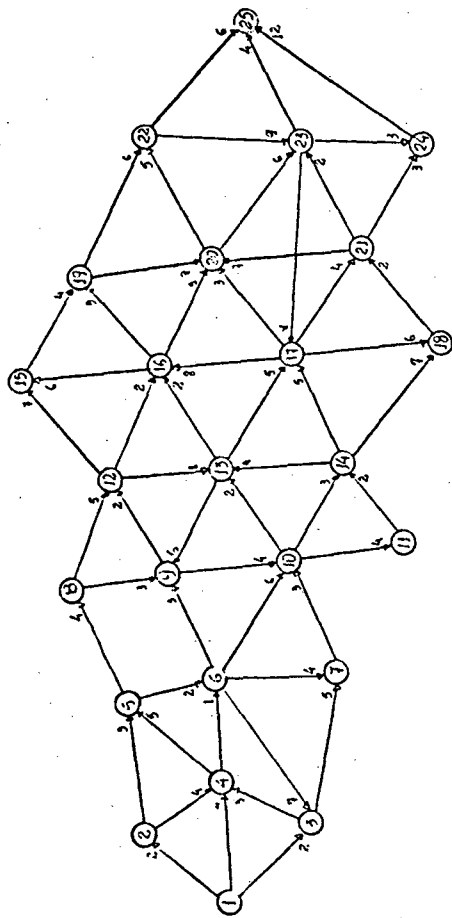
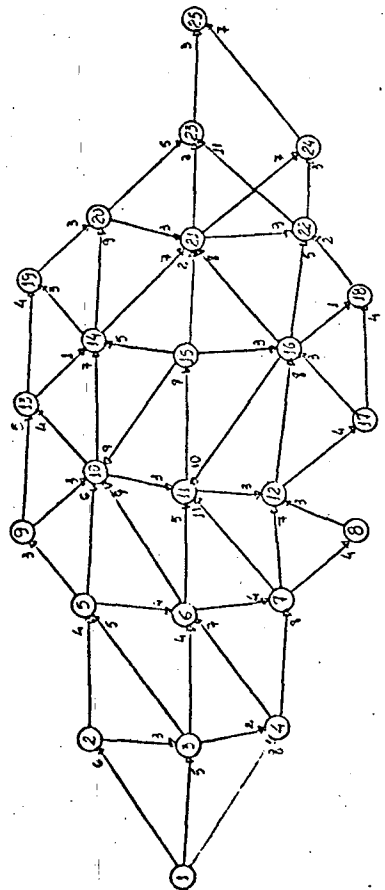
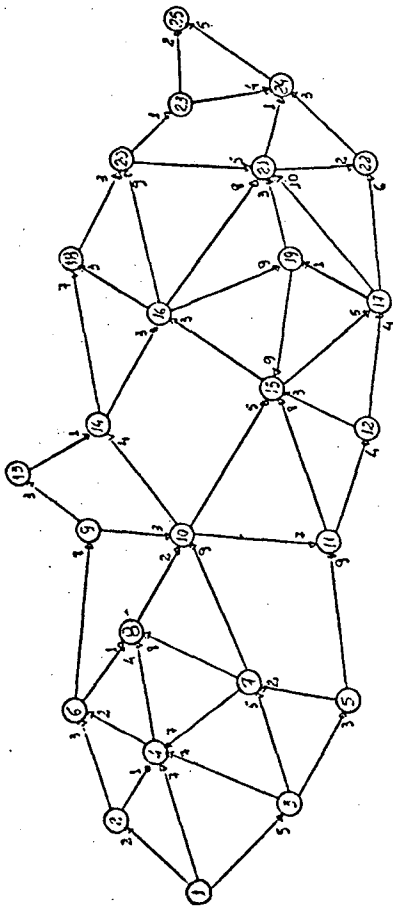
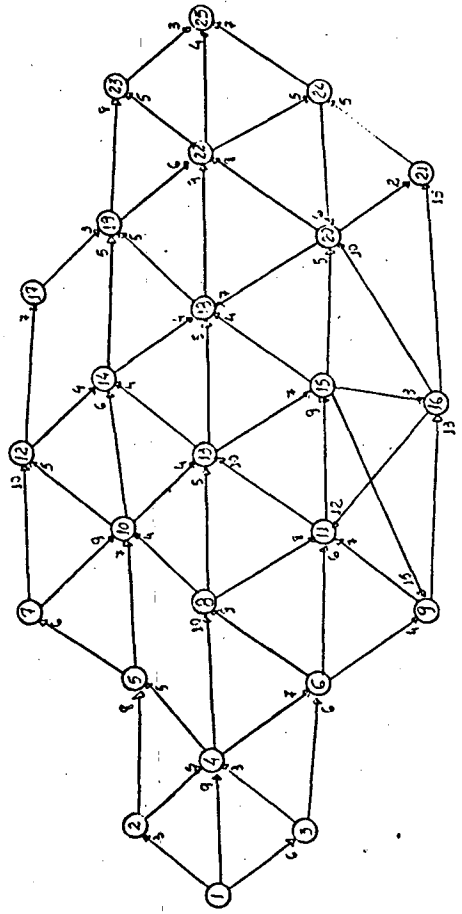
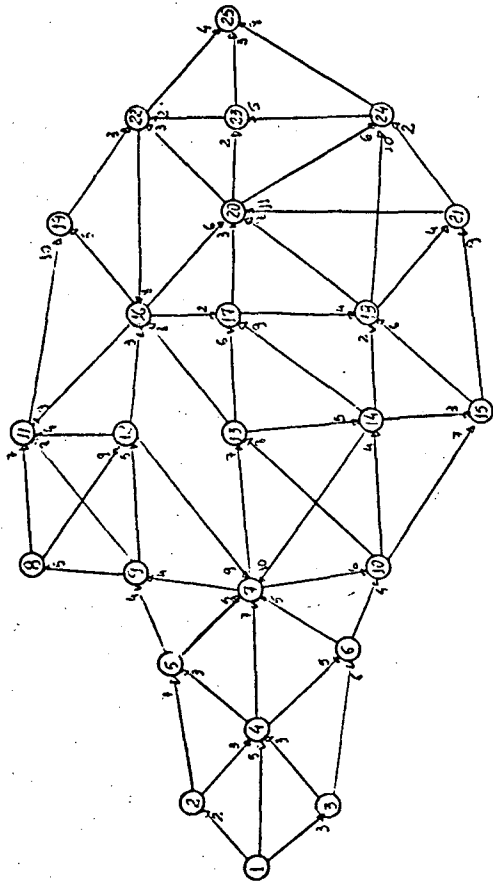
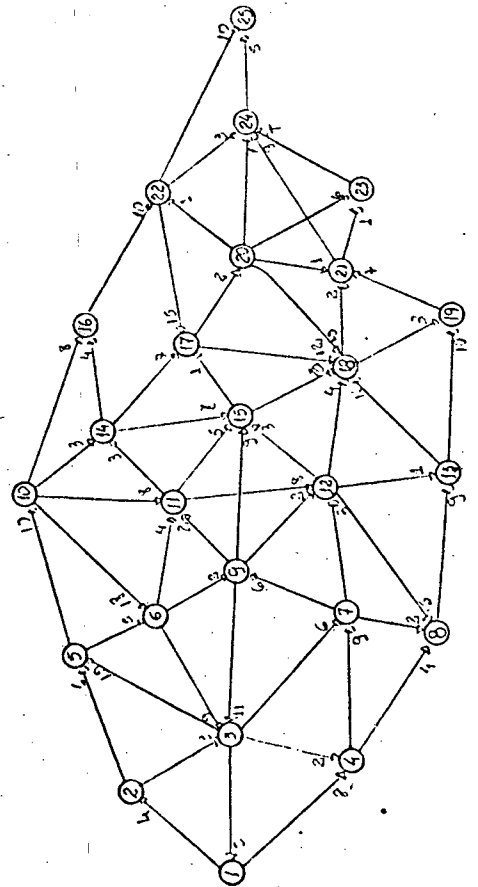
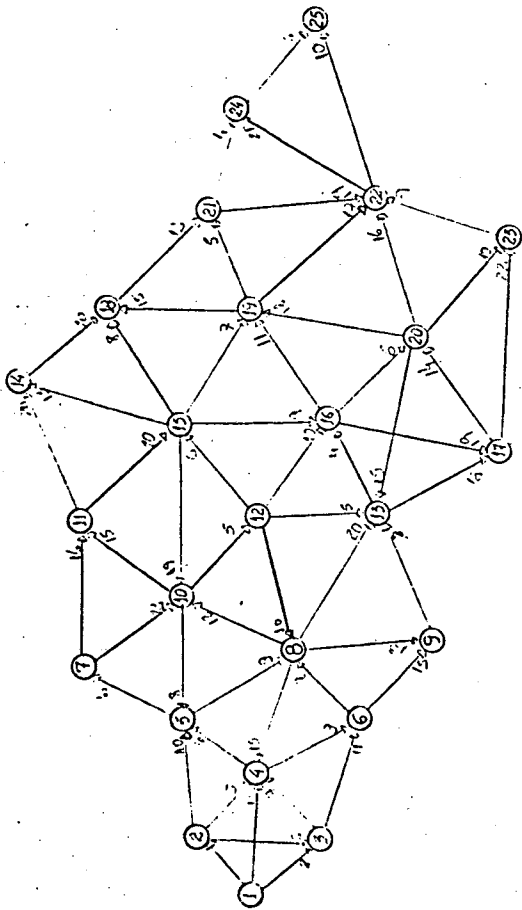
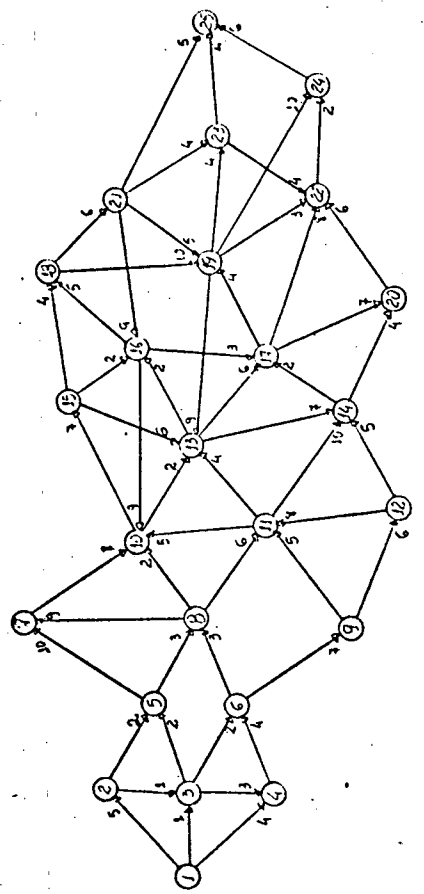
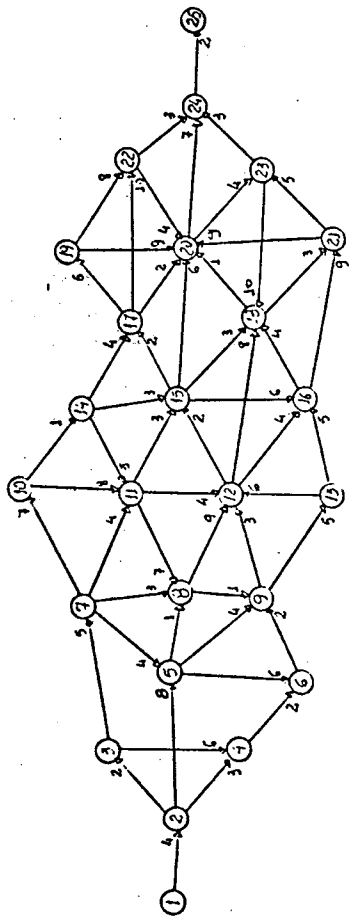


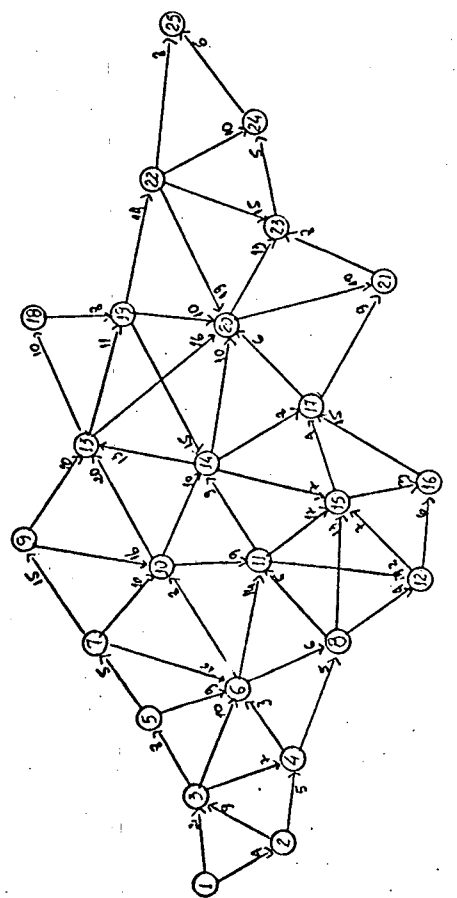
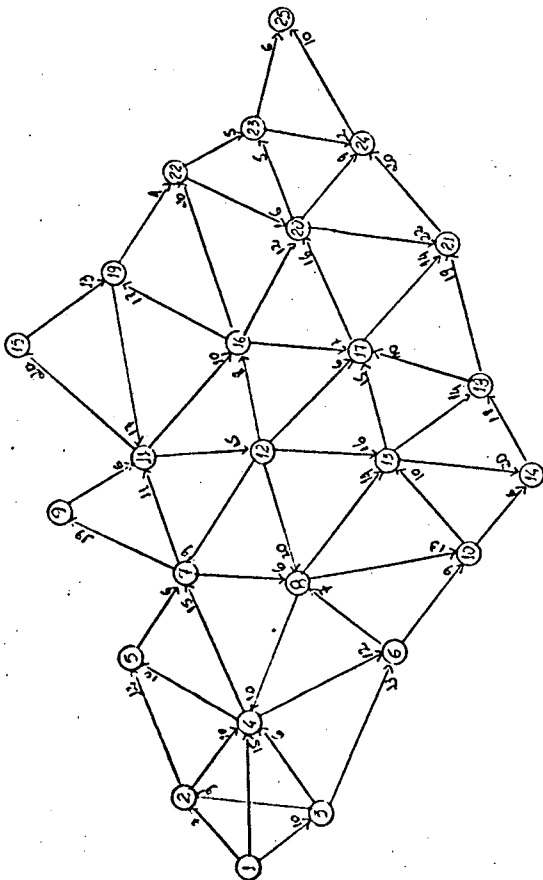
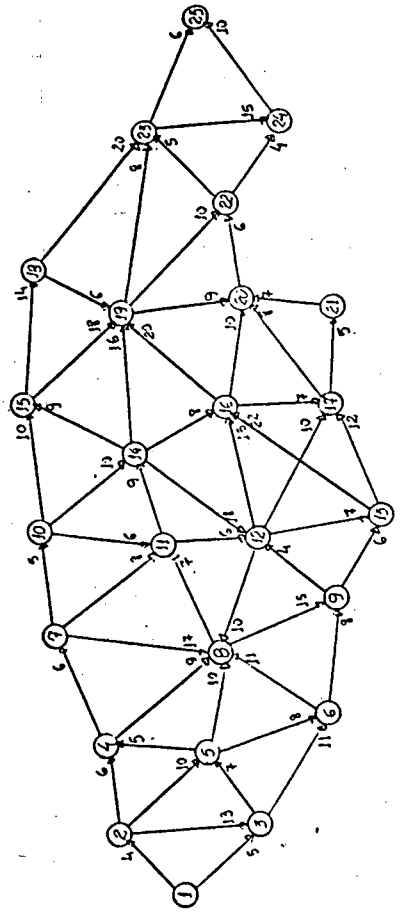
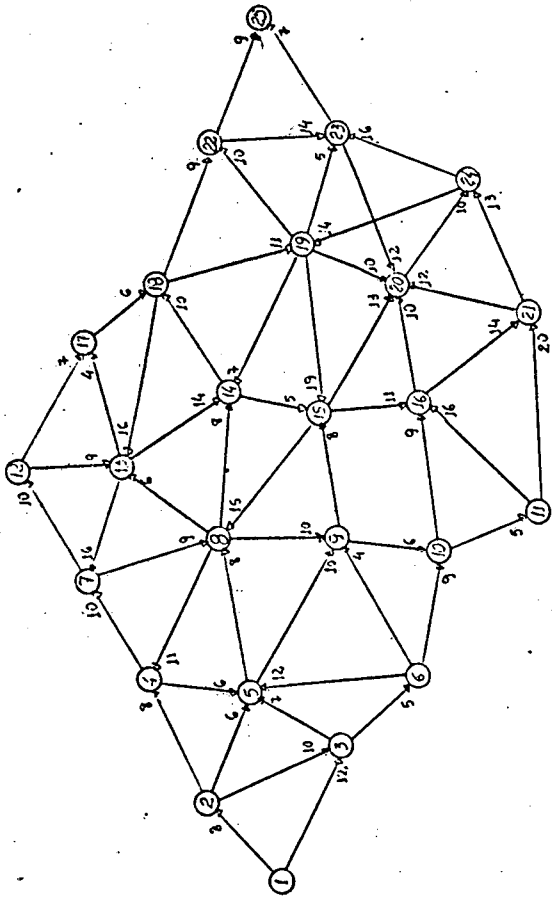
FIGURA 9

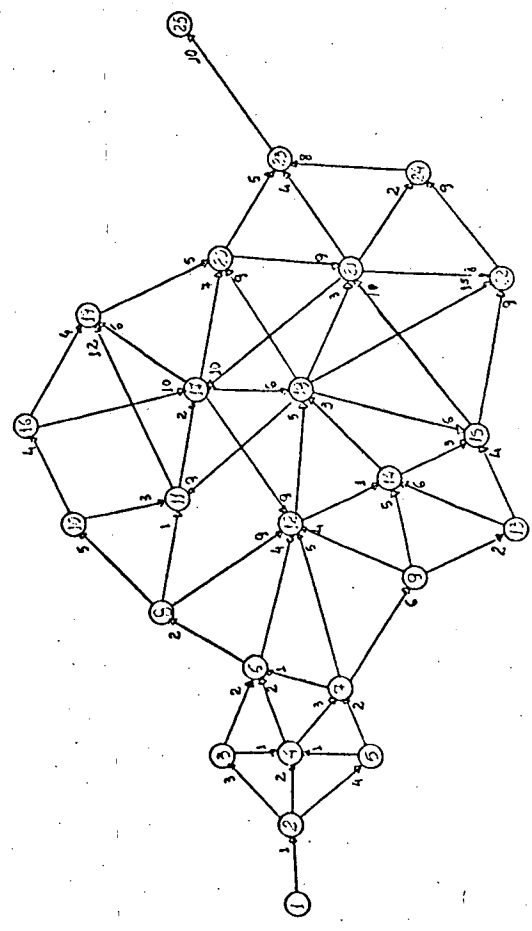
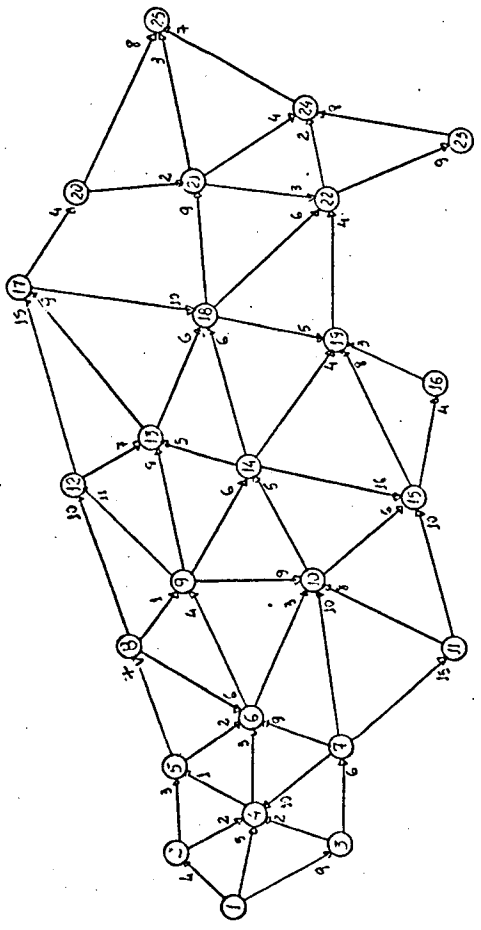
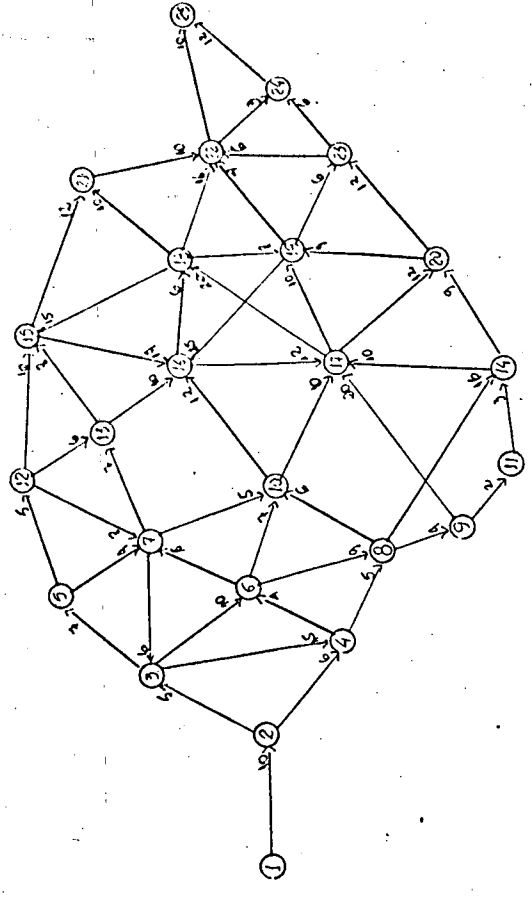
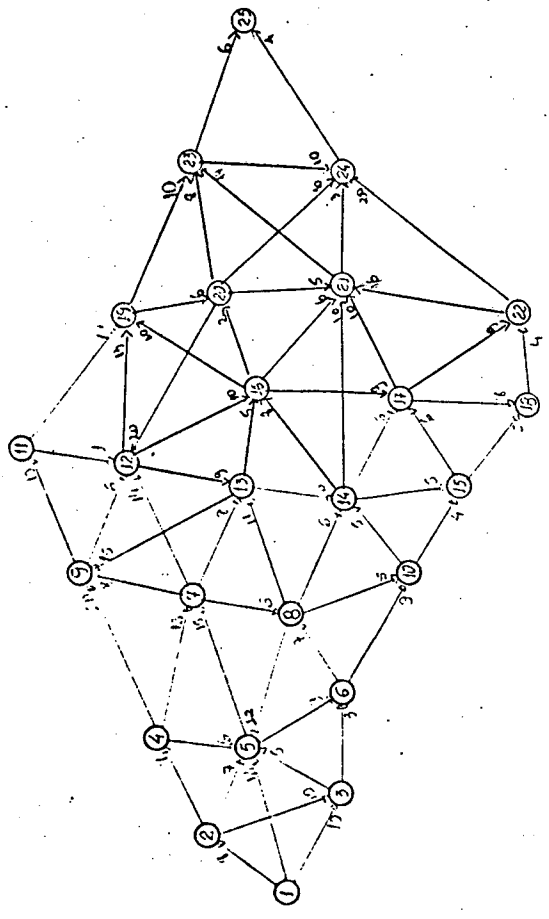
GRAFOS COM 25 VÉRTICES





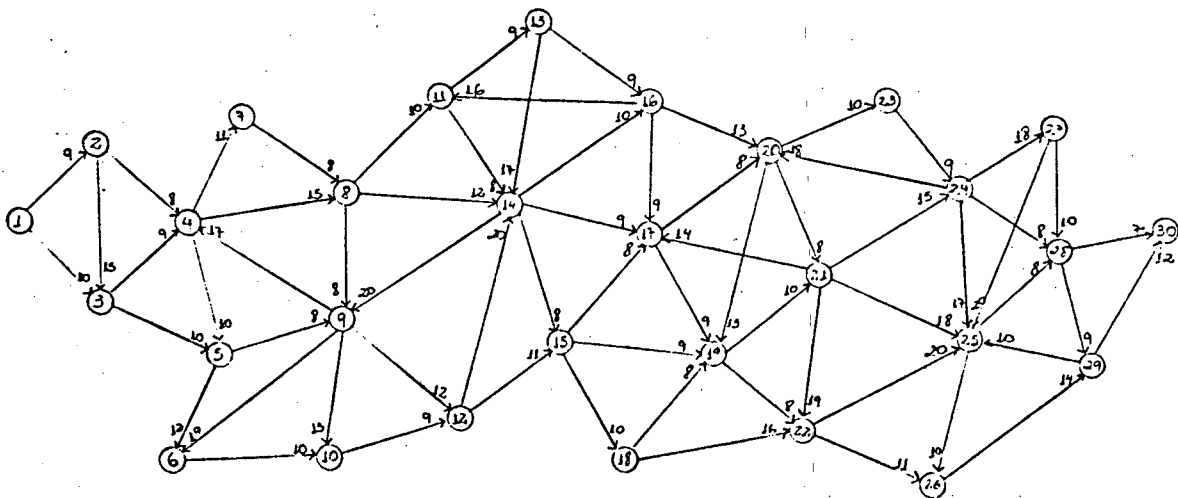
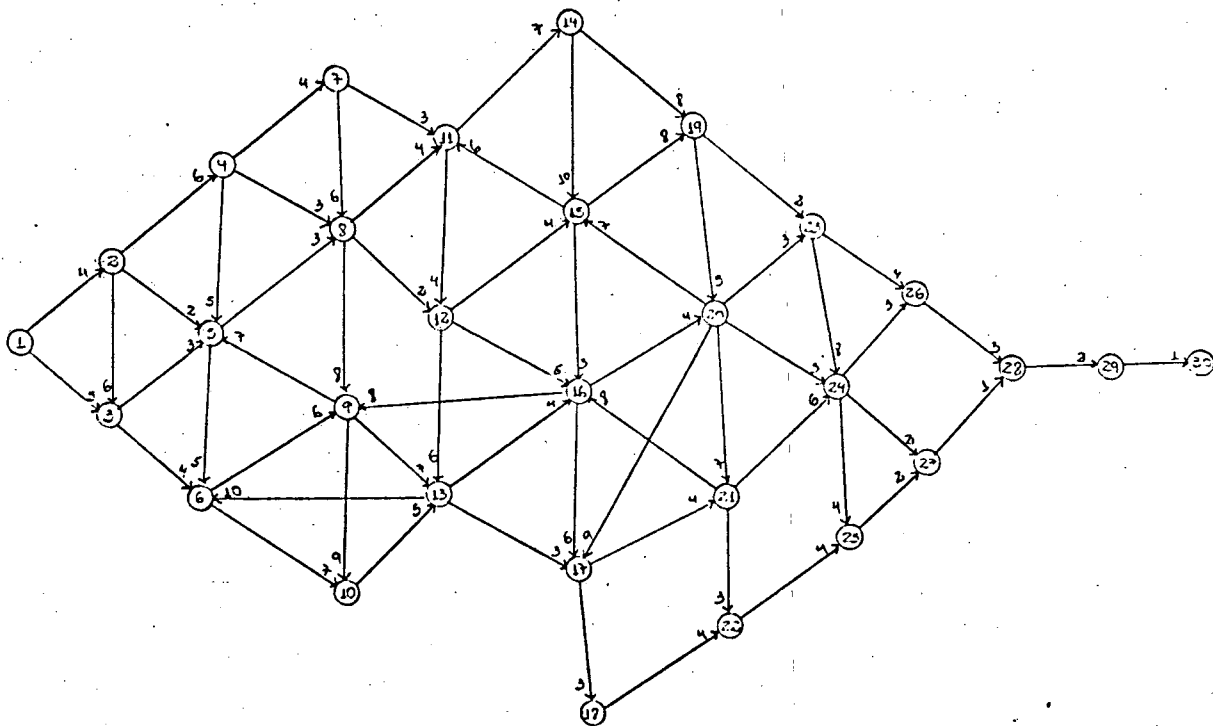


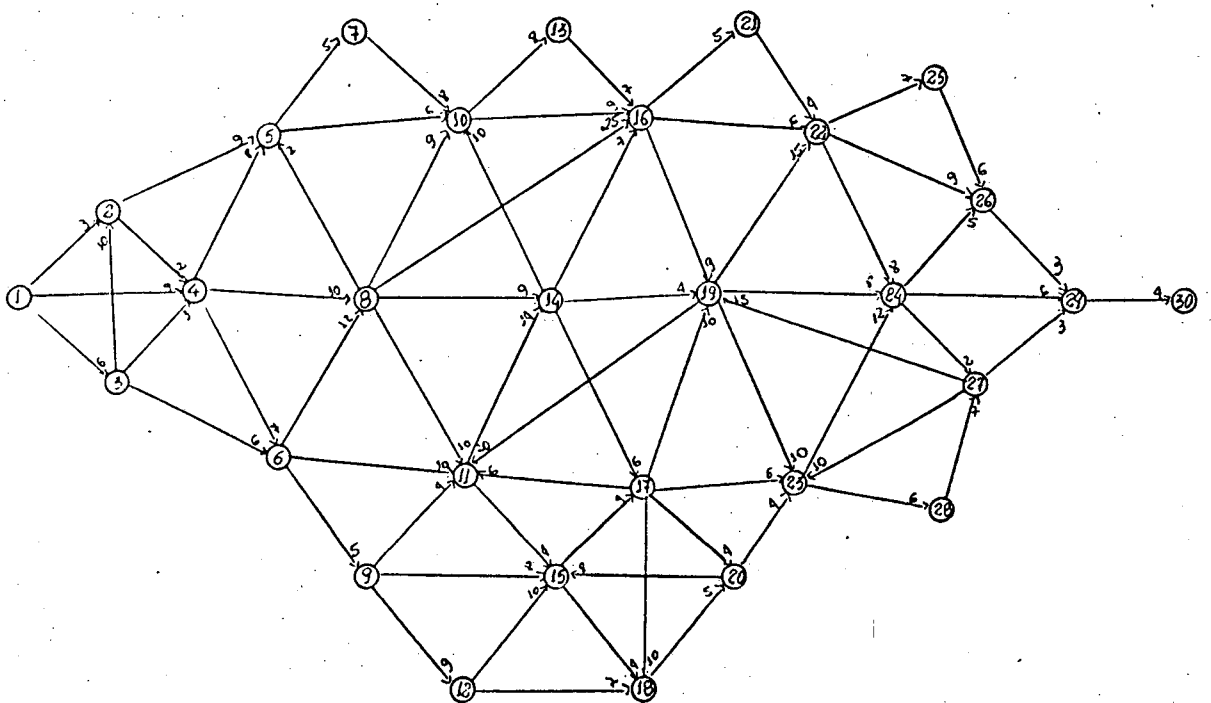
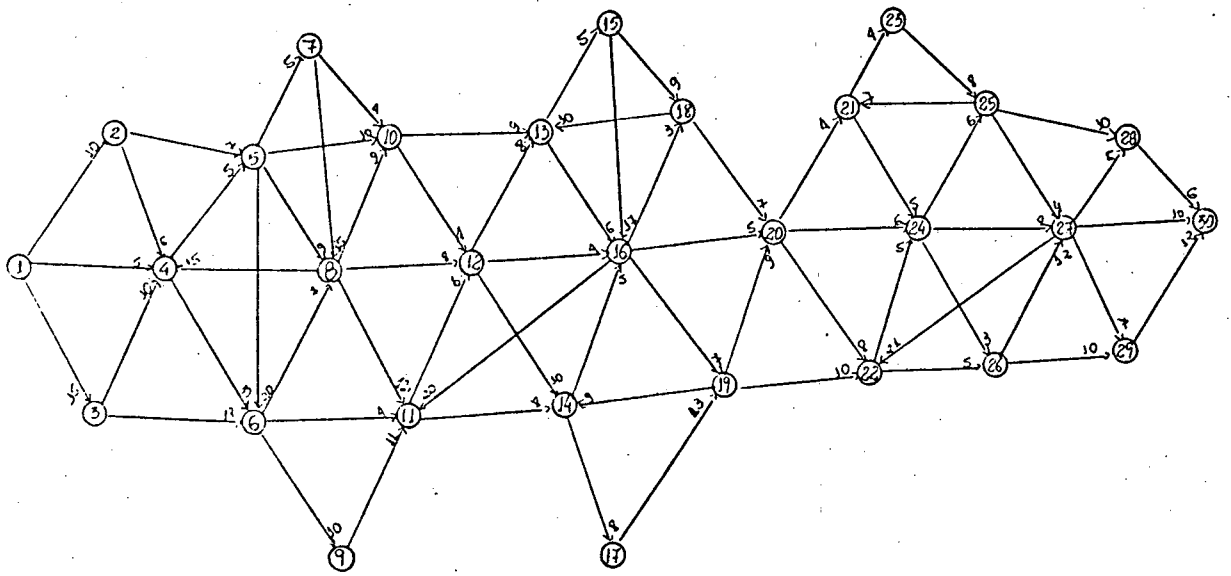


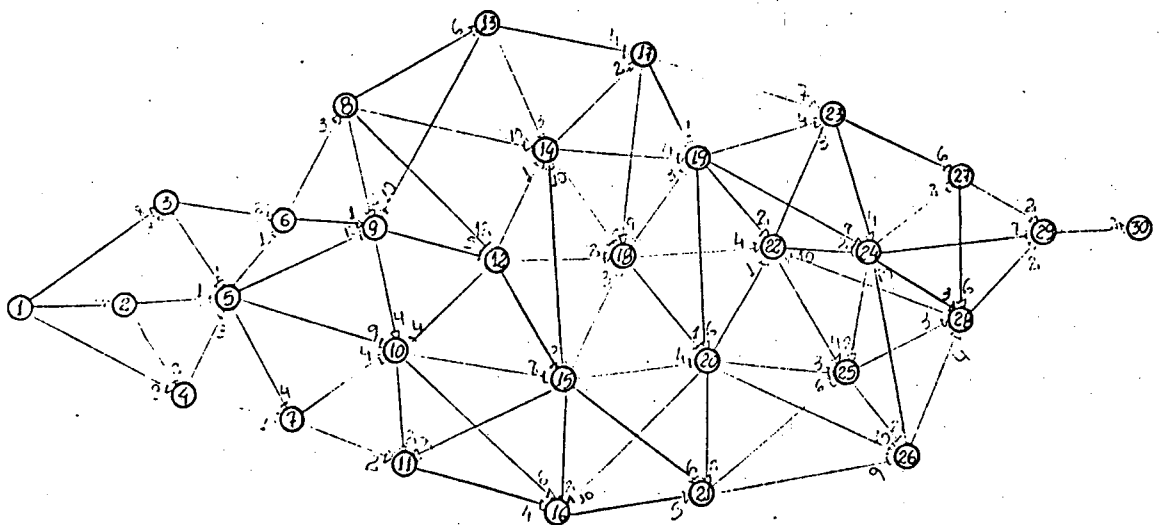
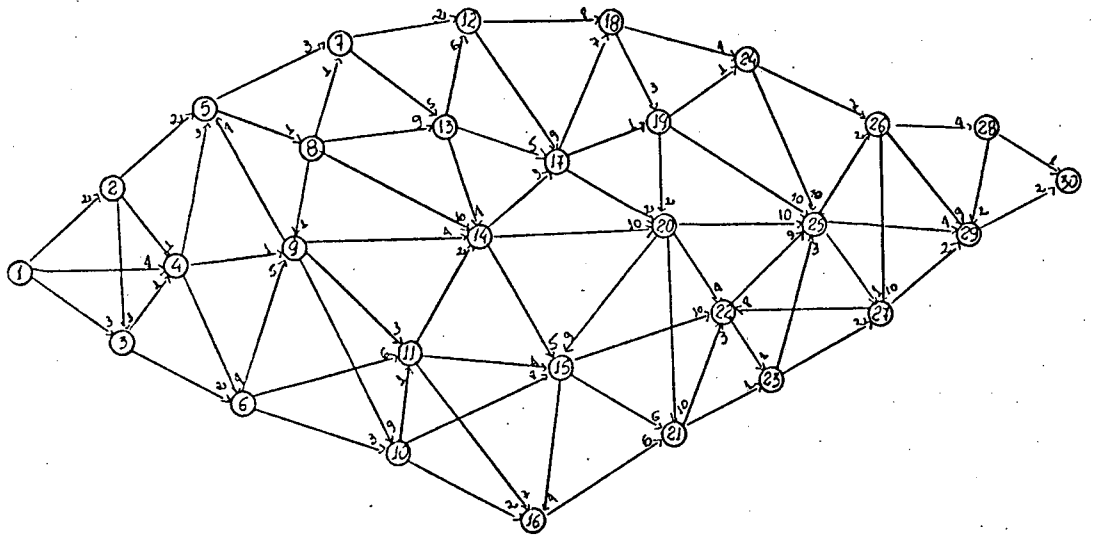


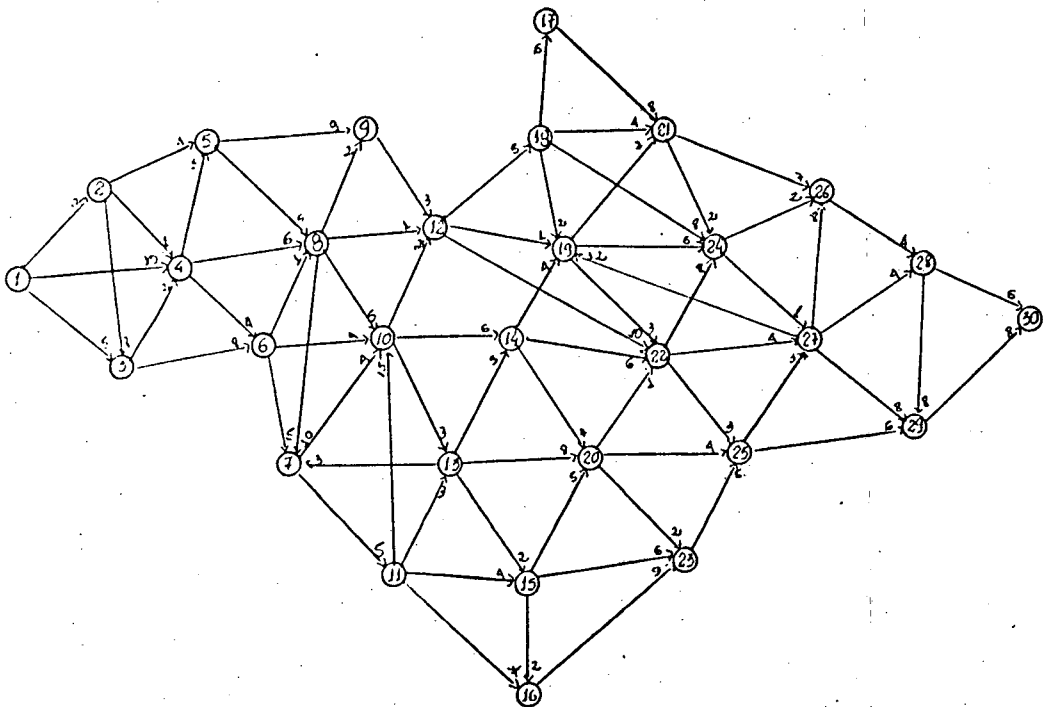
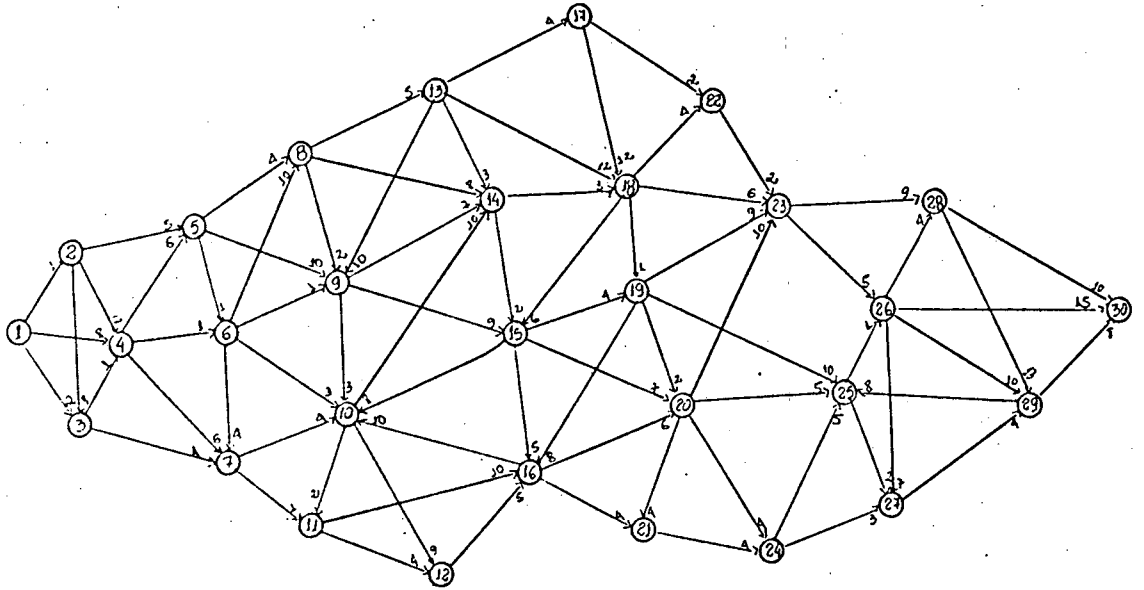
F I G U R A 10

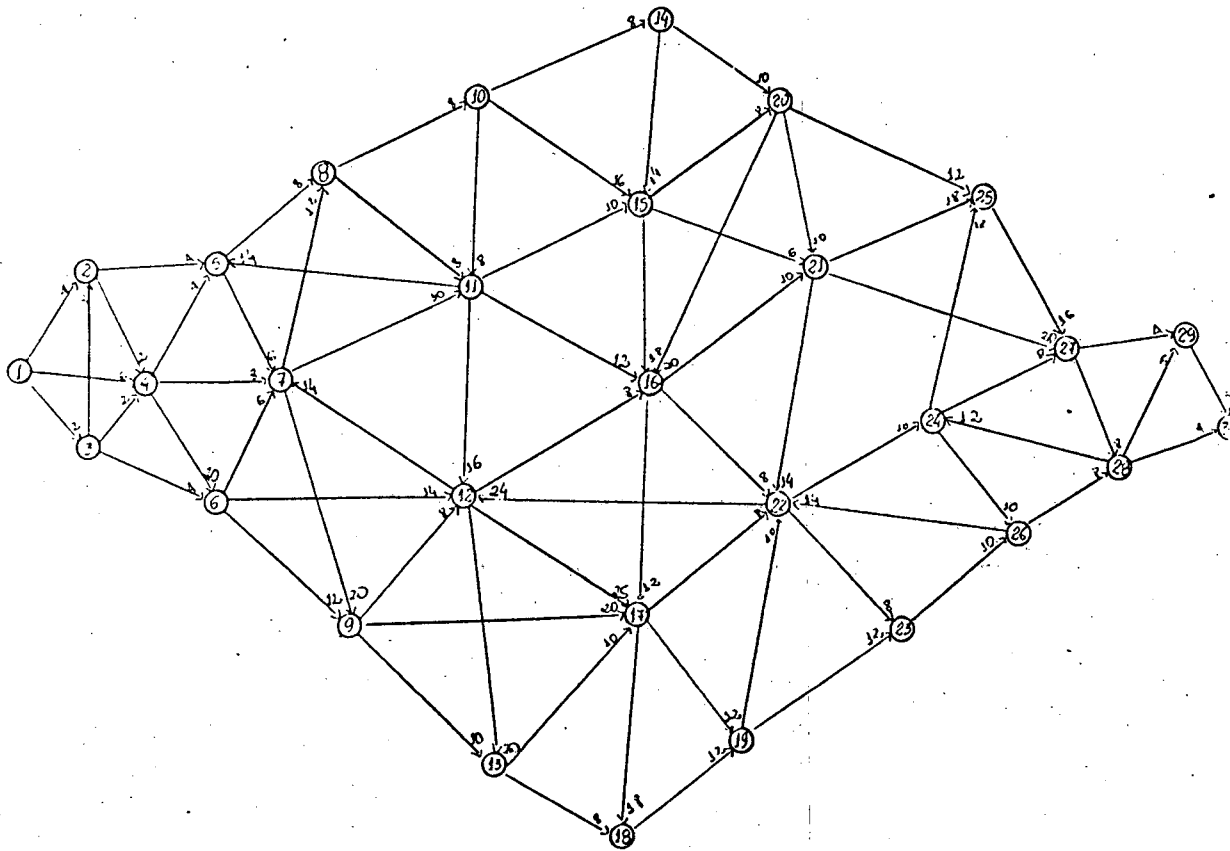
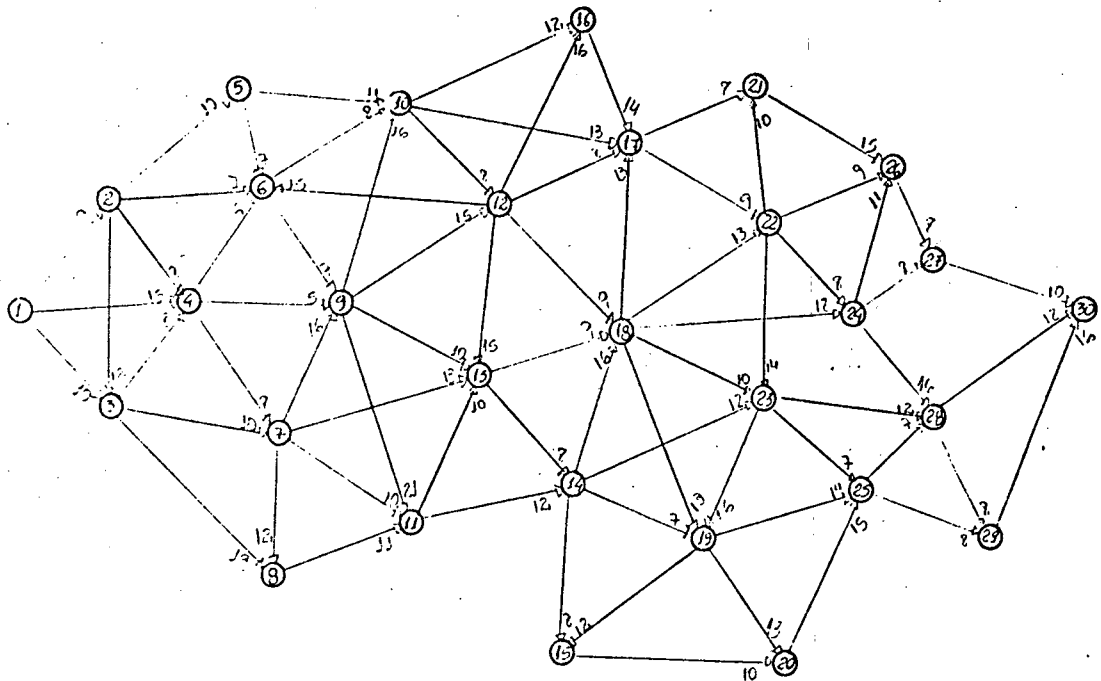
GRAFOS COM 30 VÉRTICES

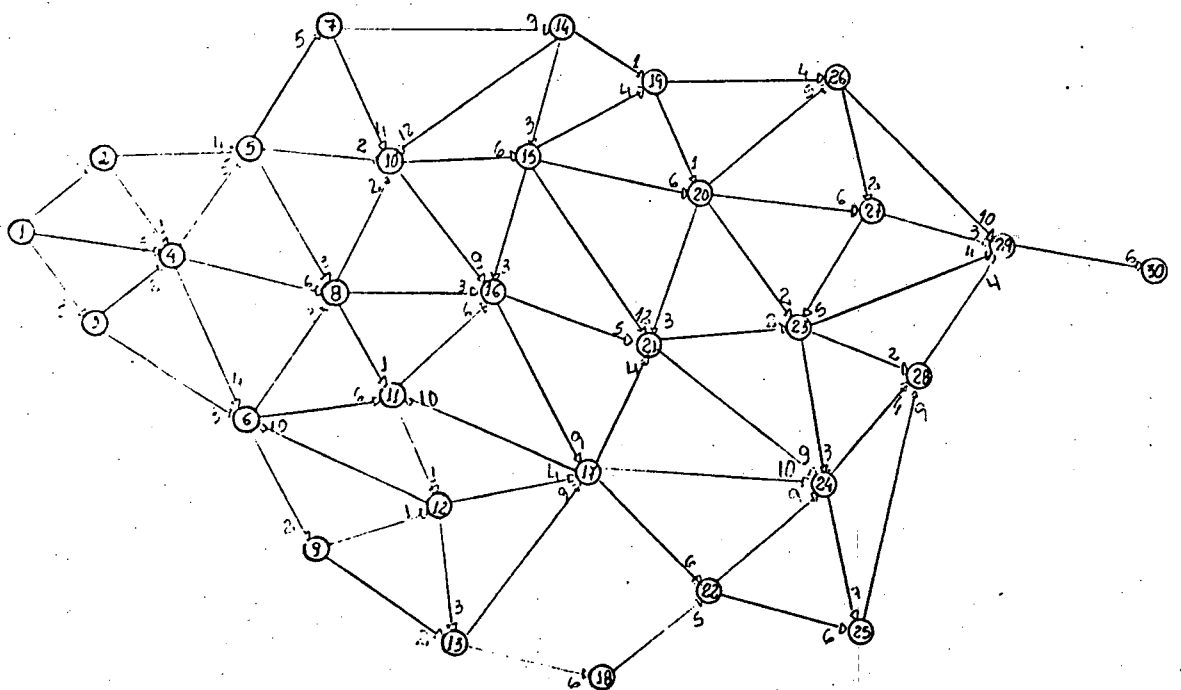
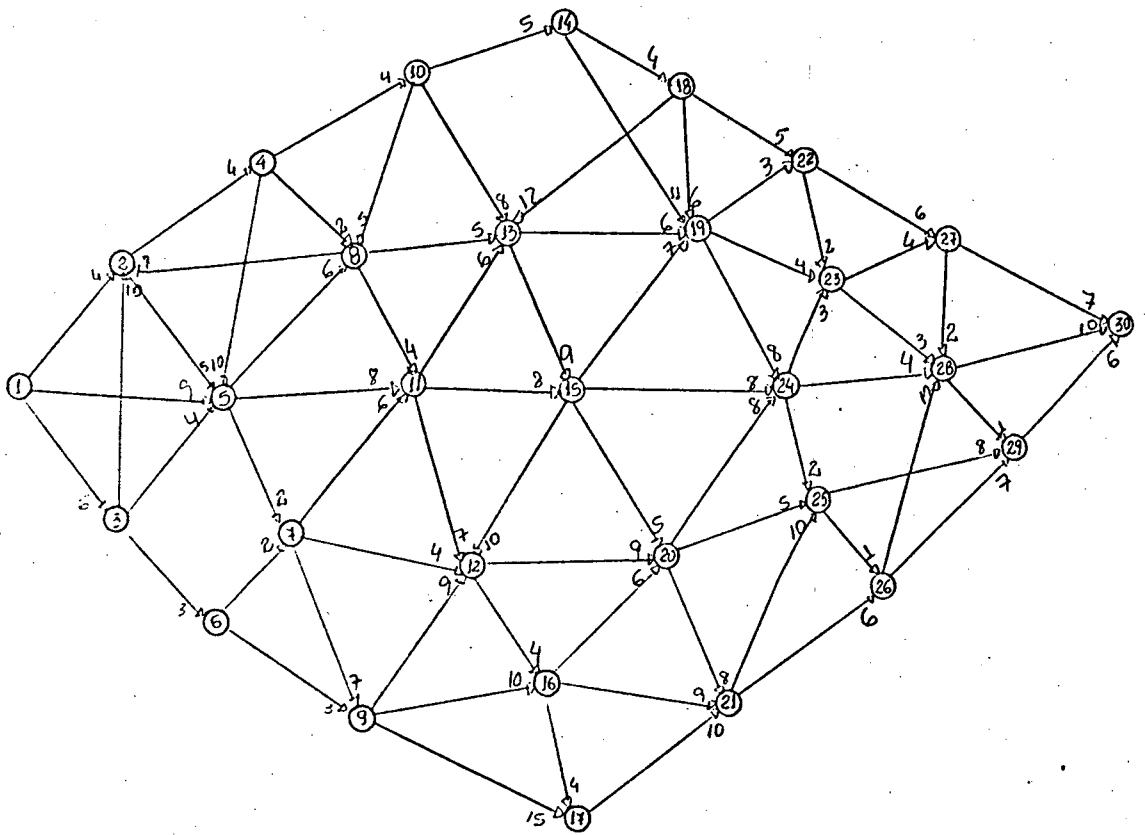


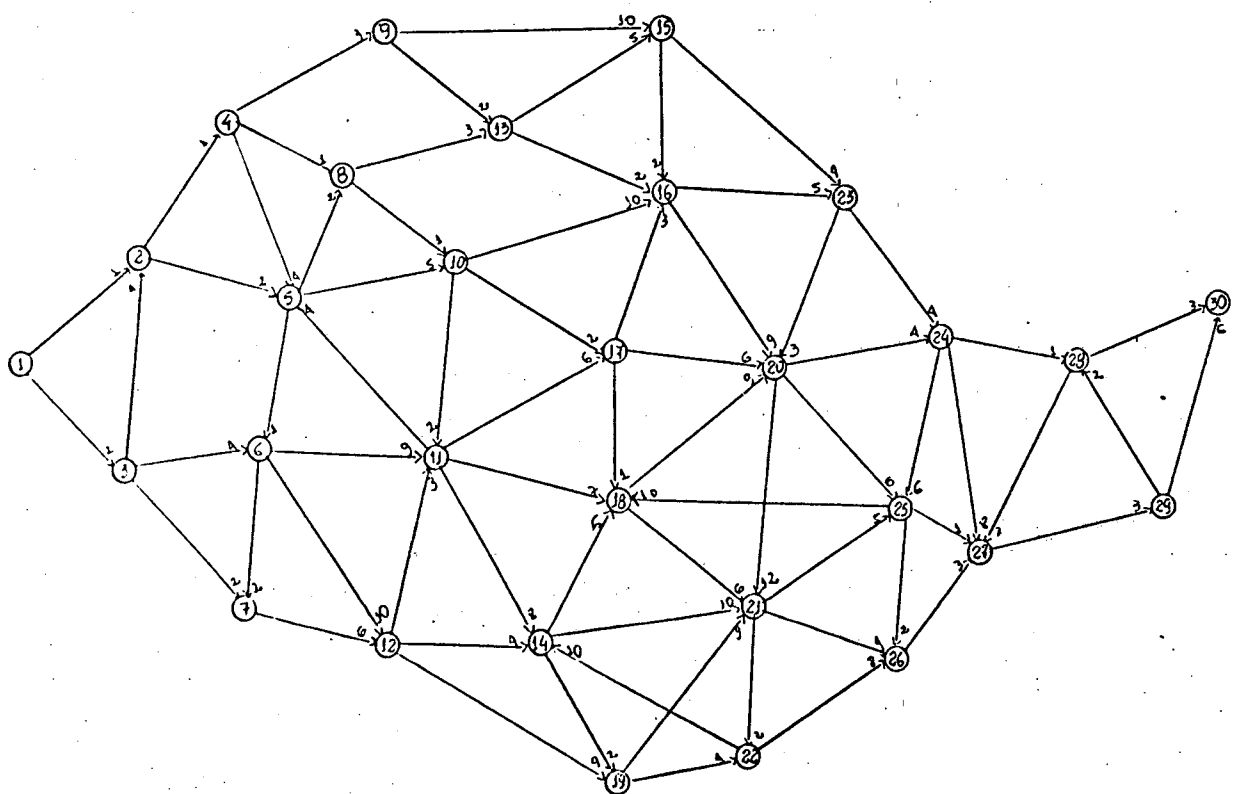
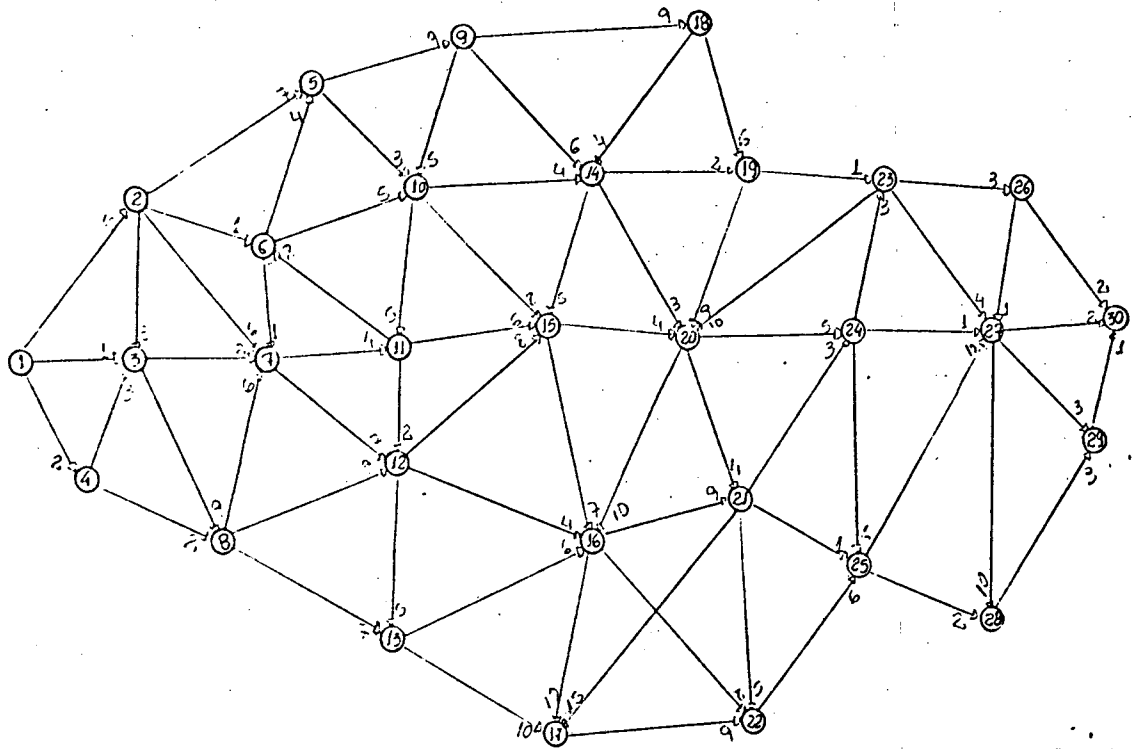


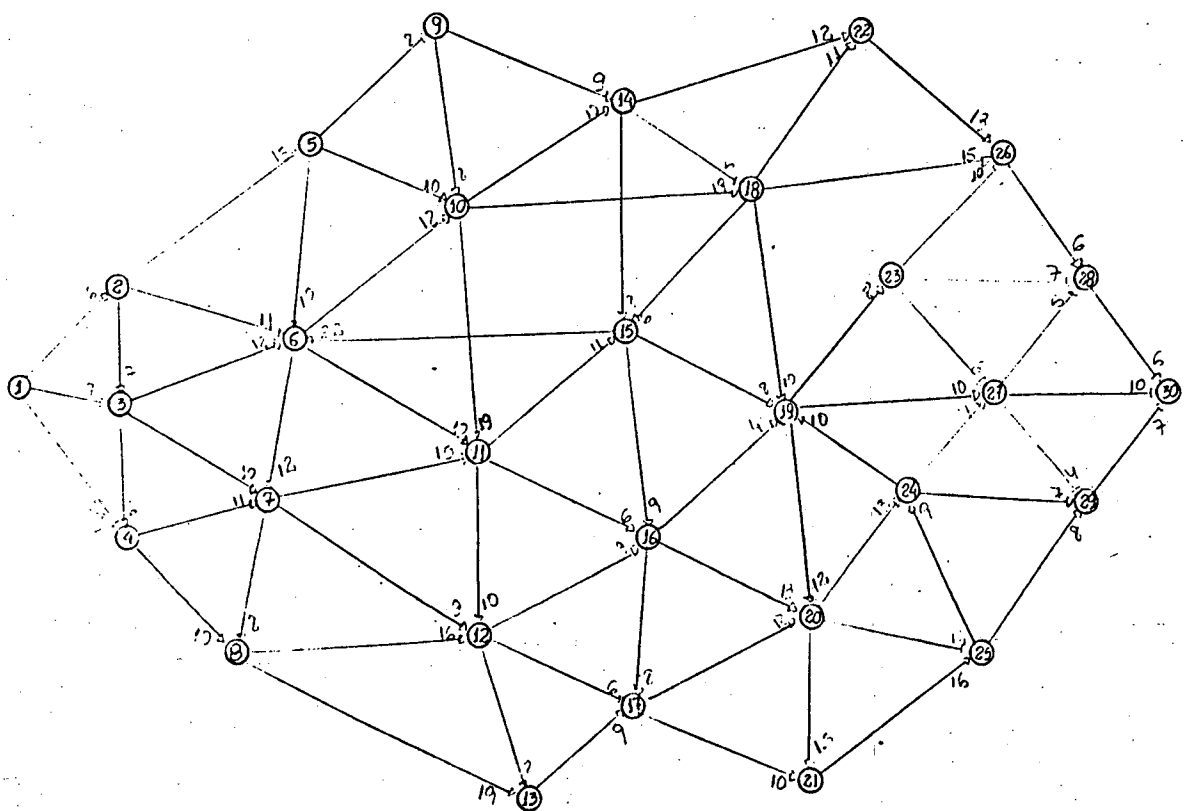
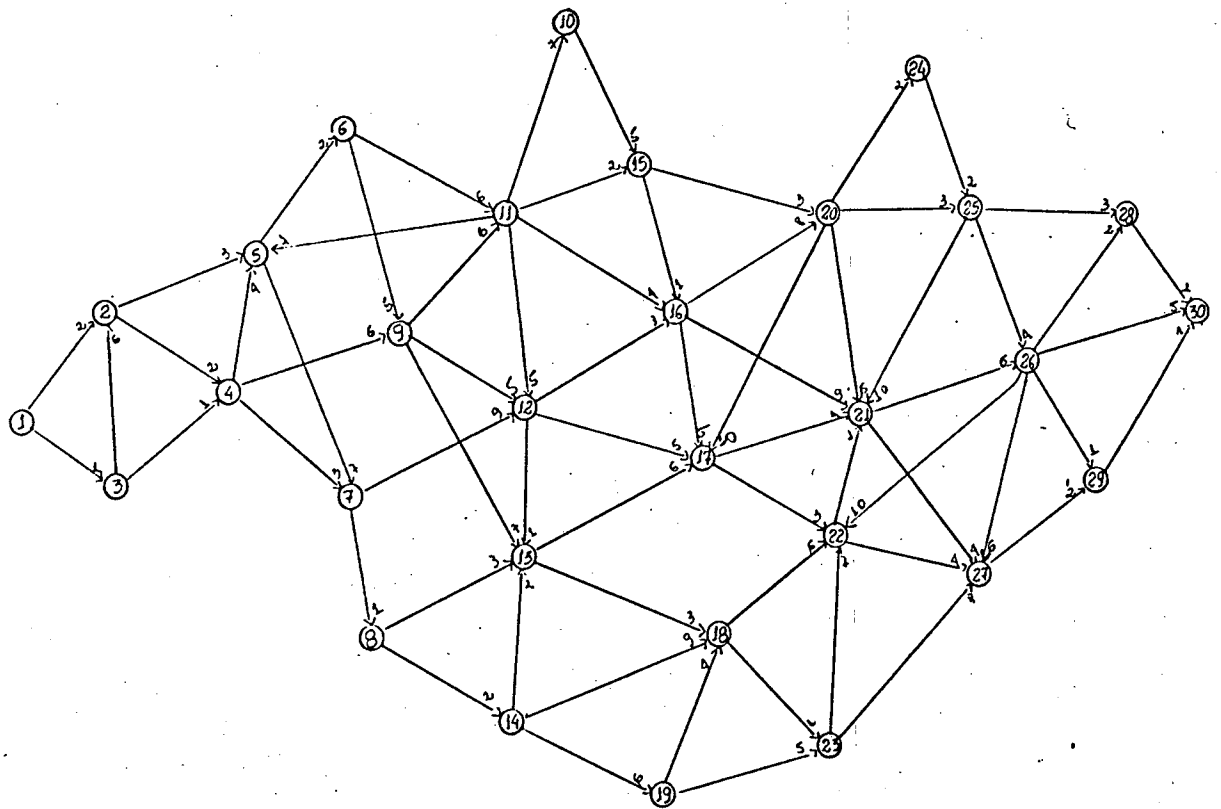


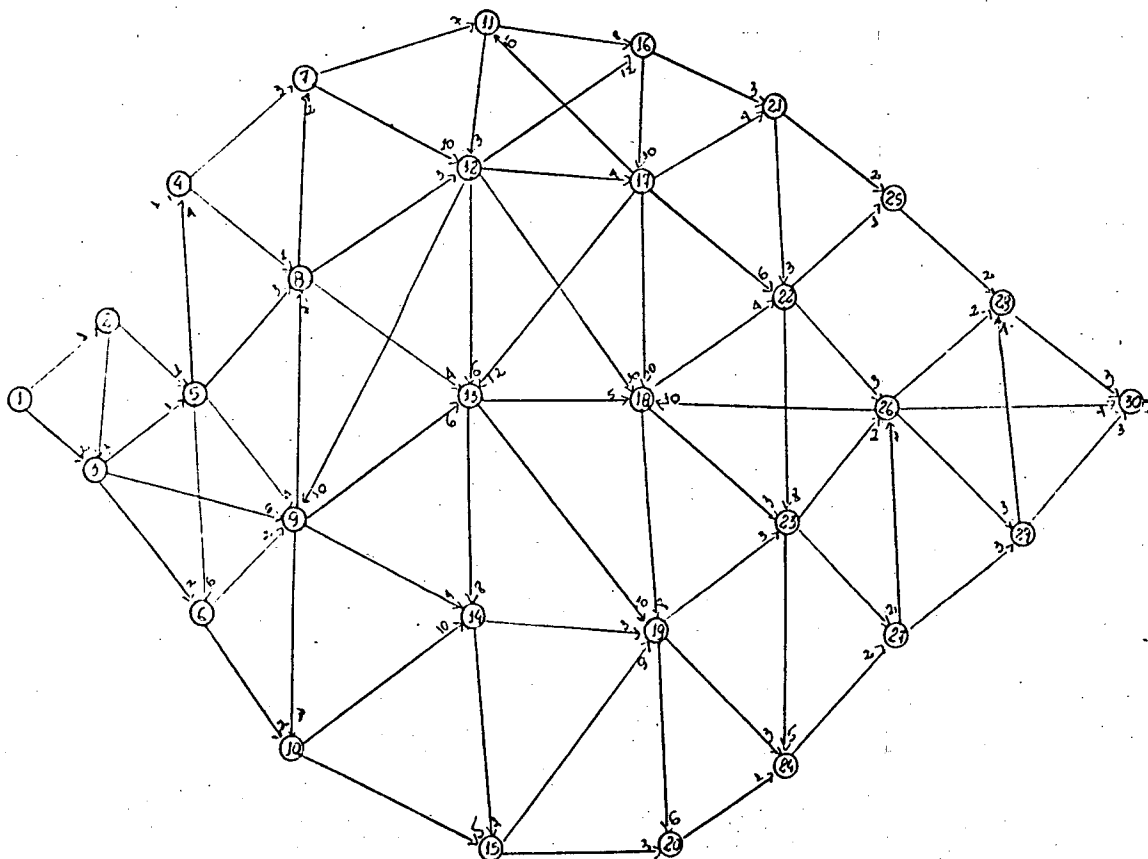
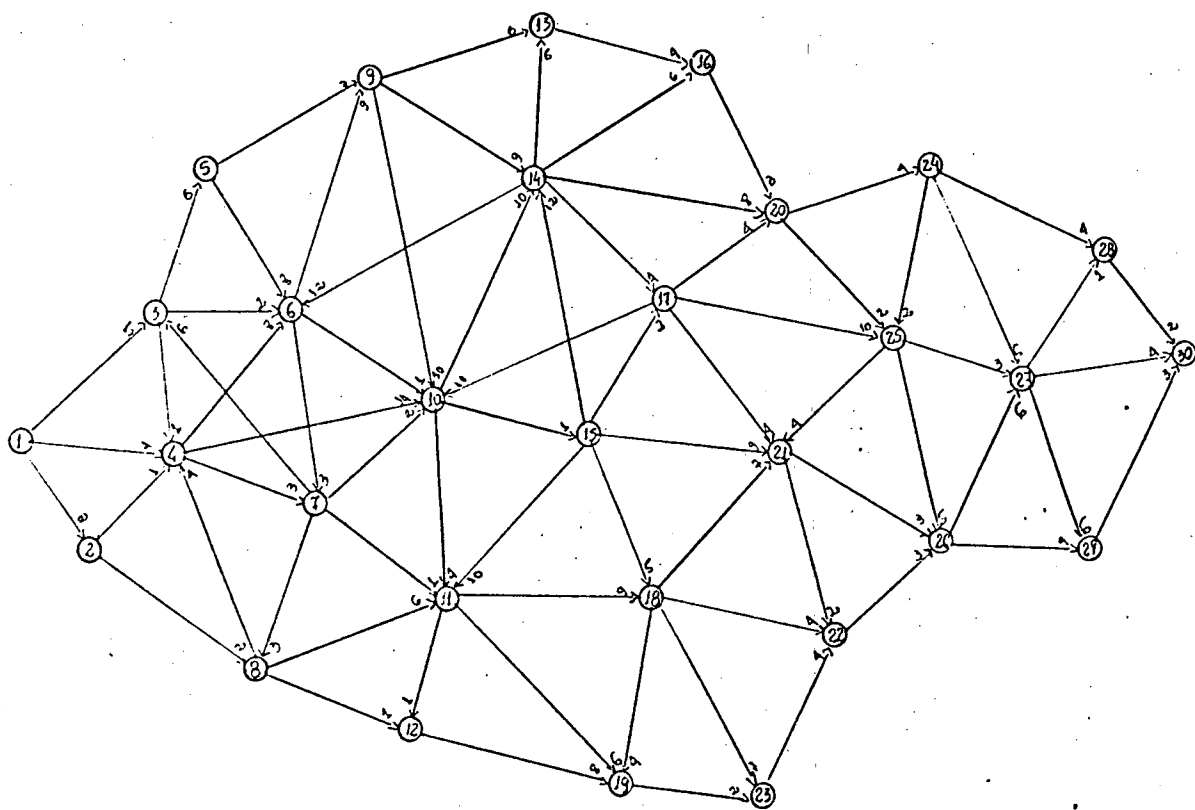






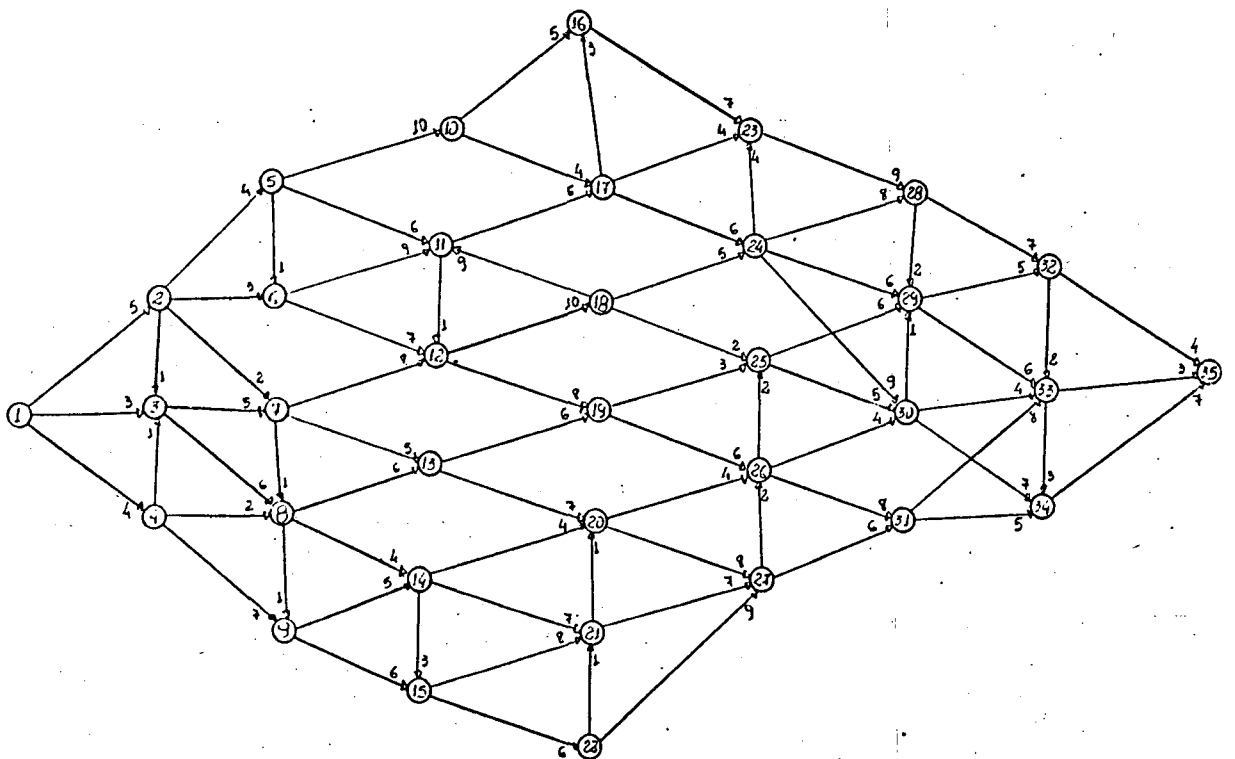
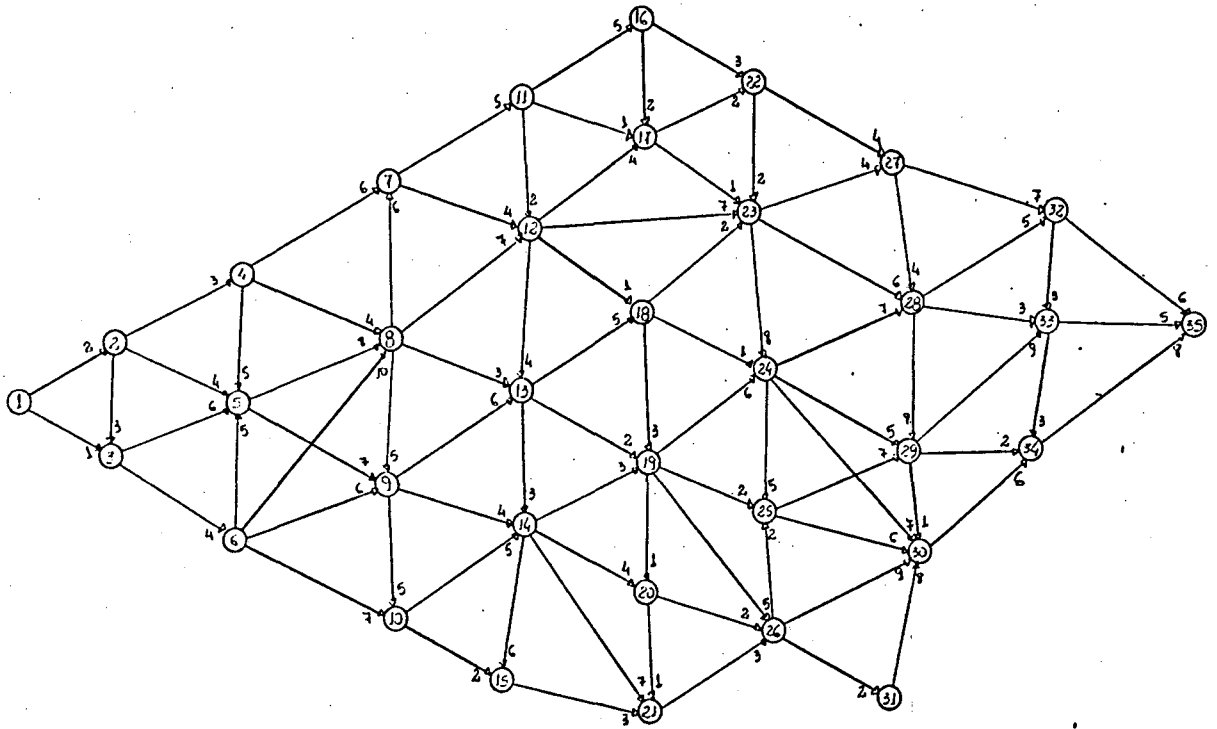


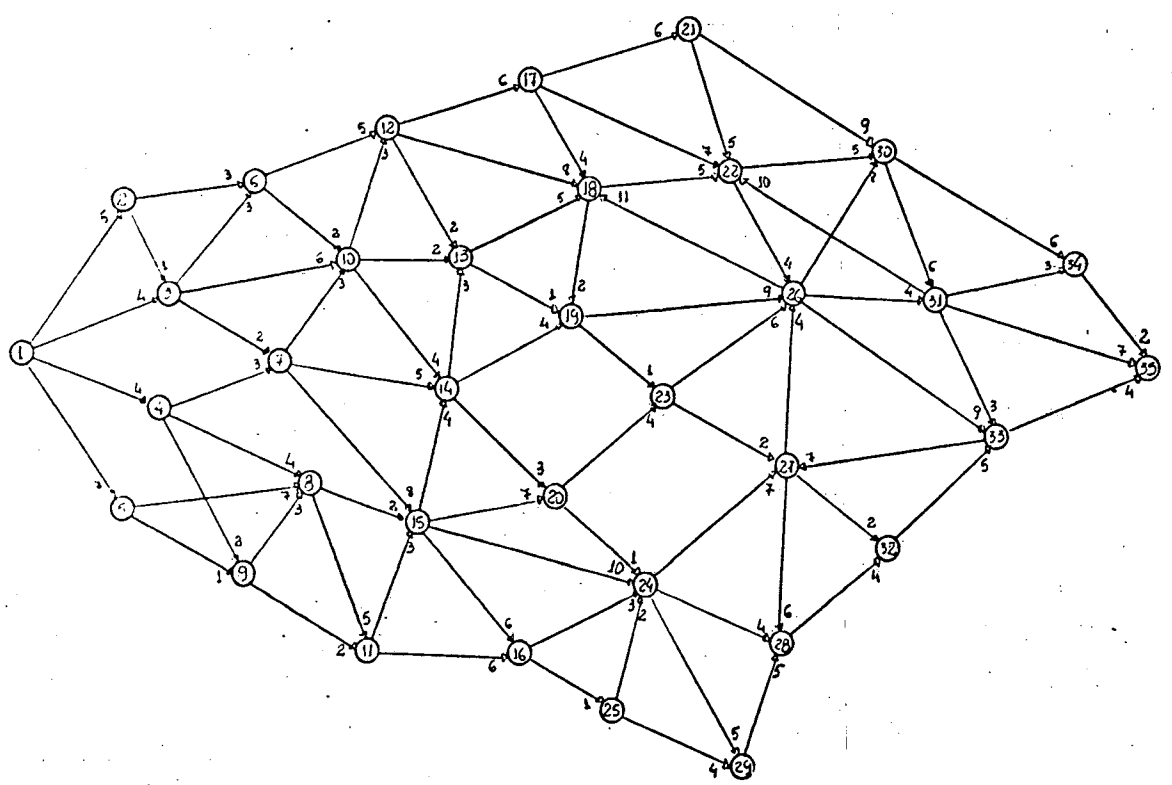
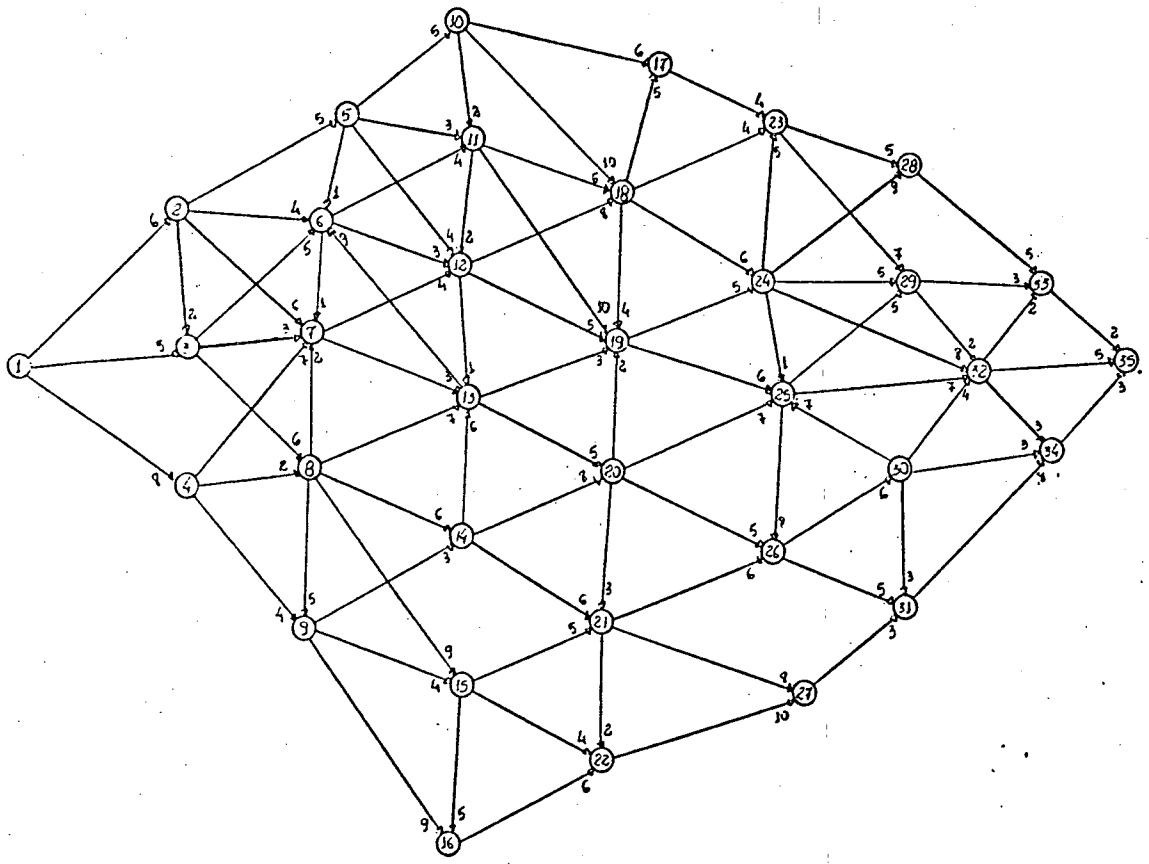


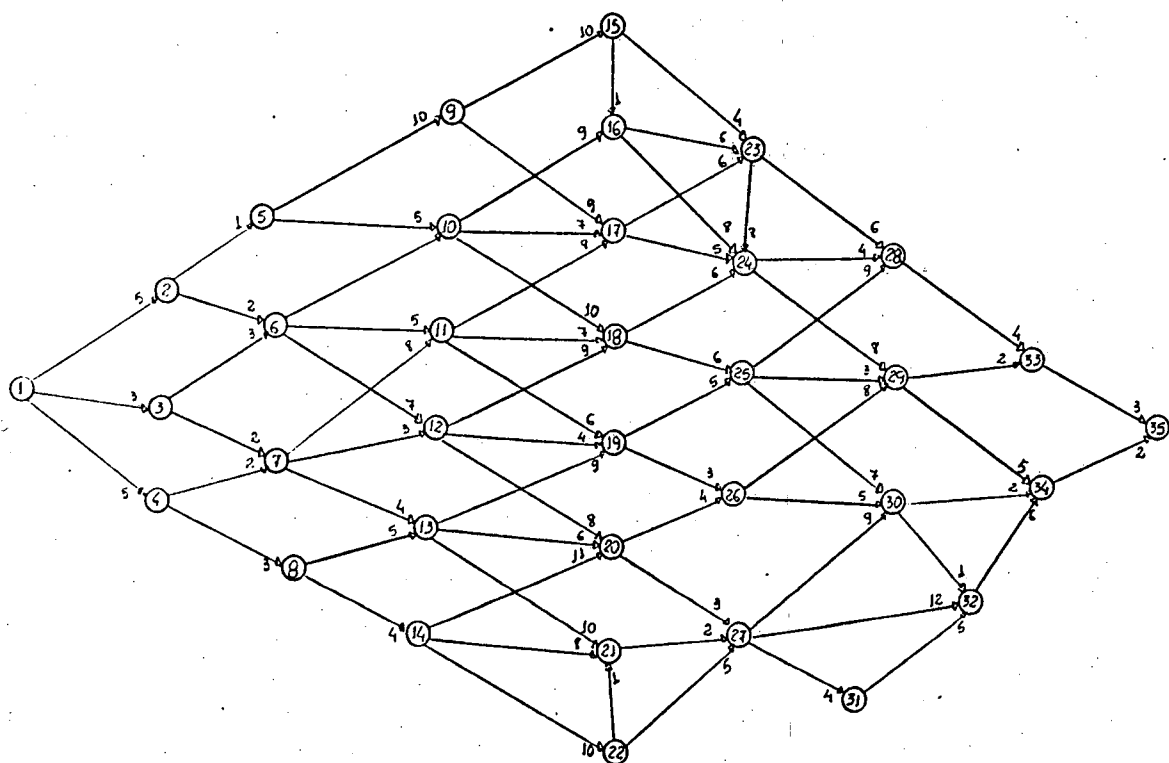
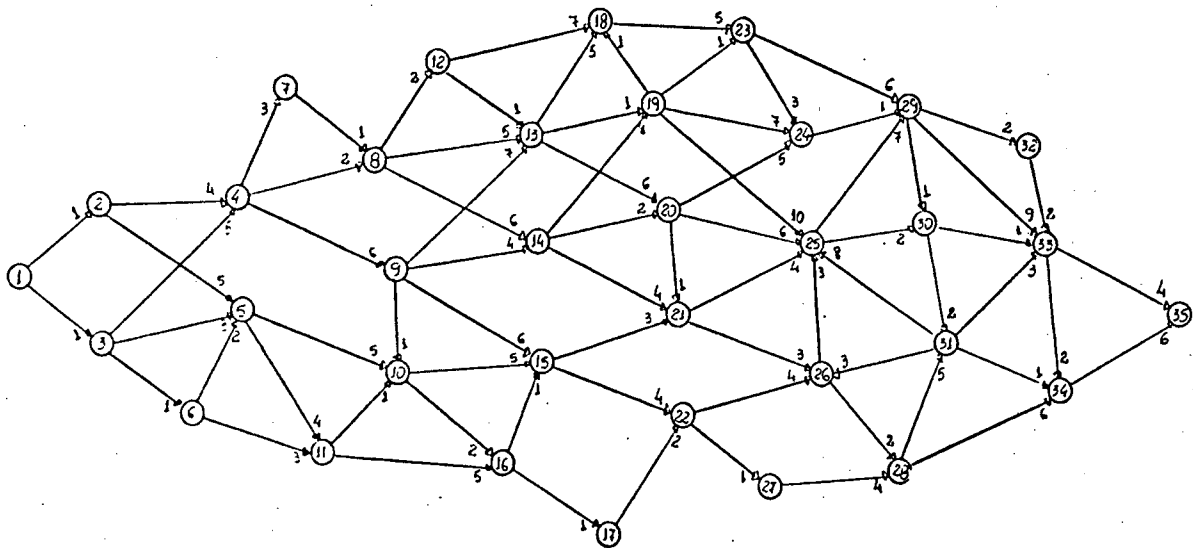


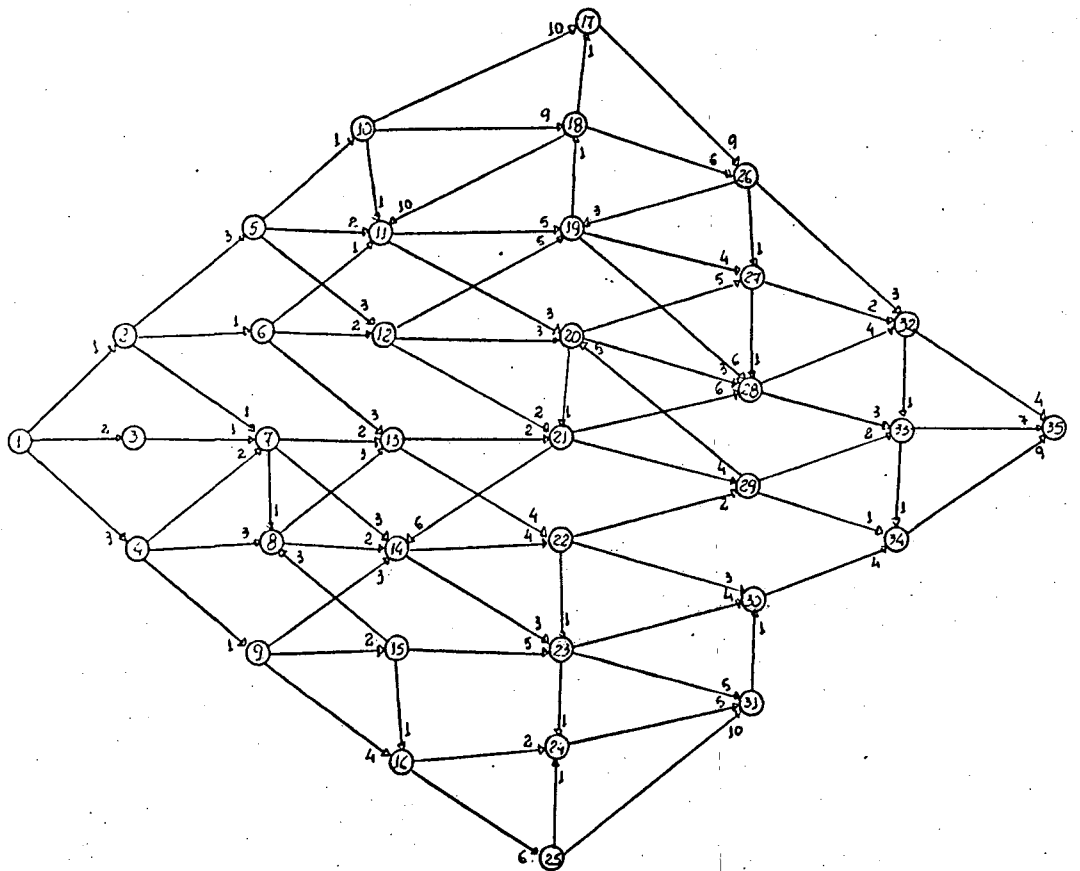
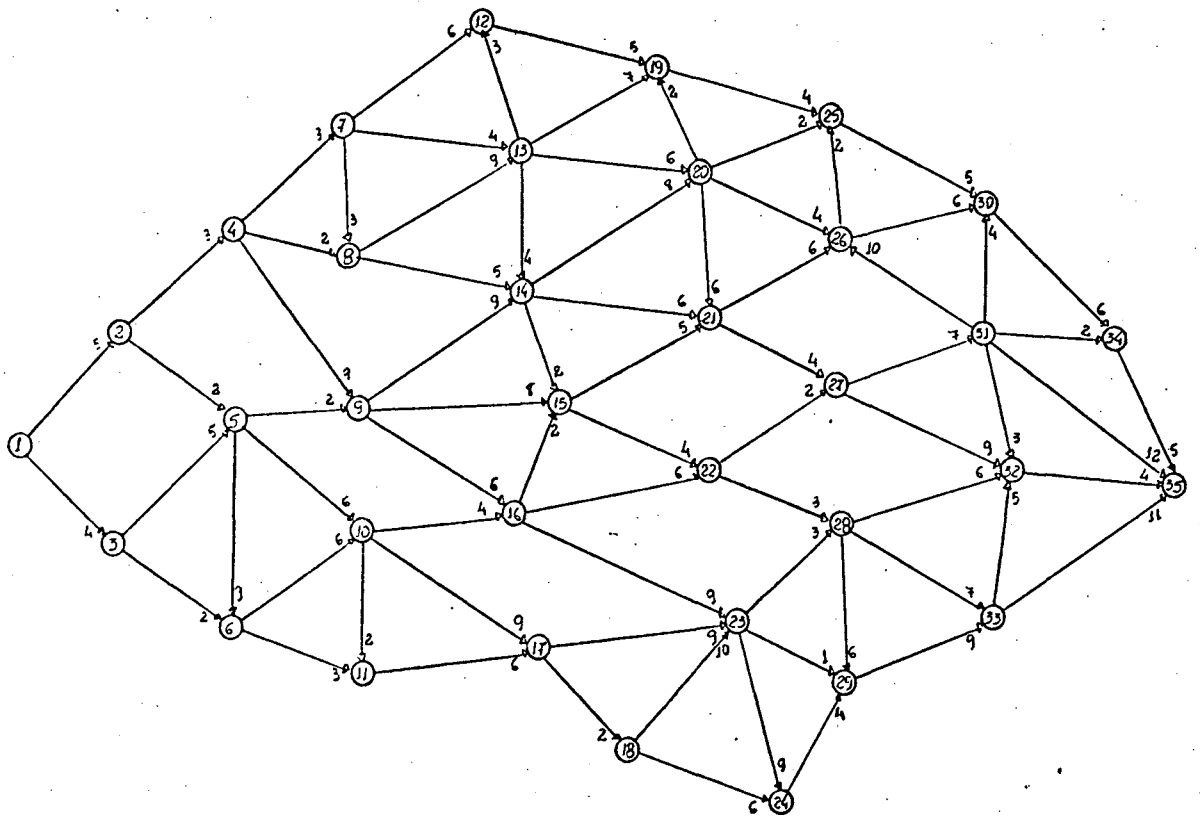
F I G U R A 11

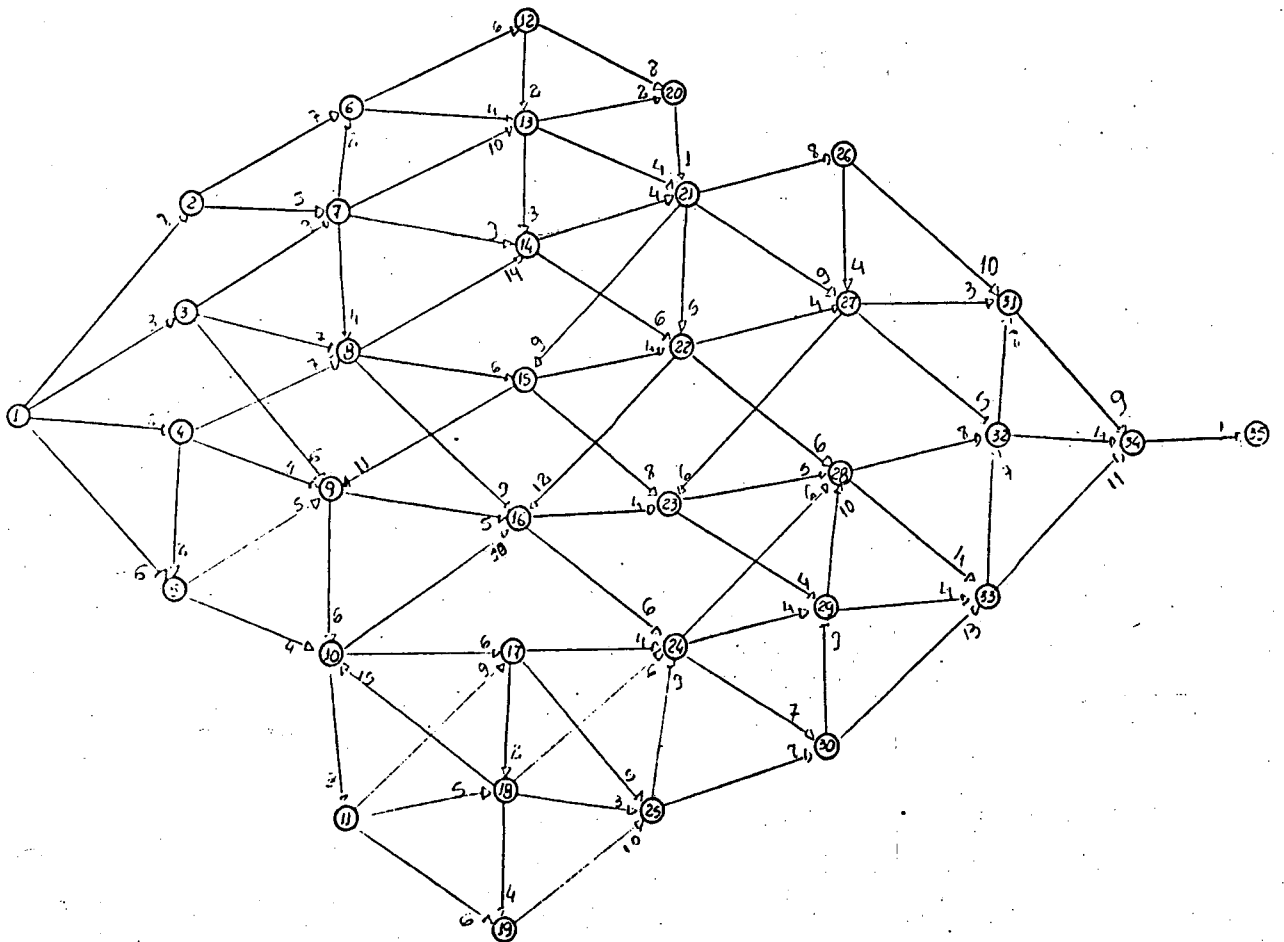
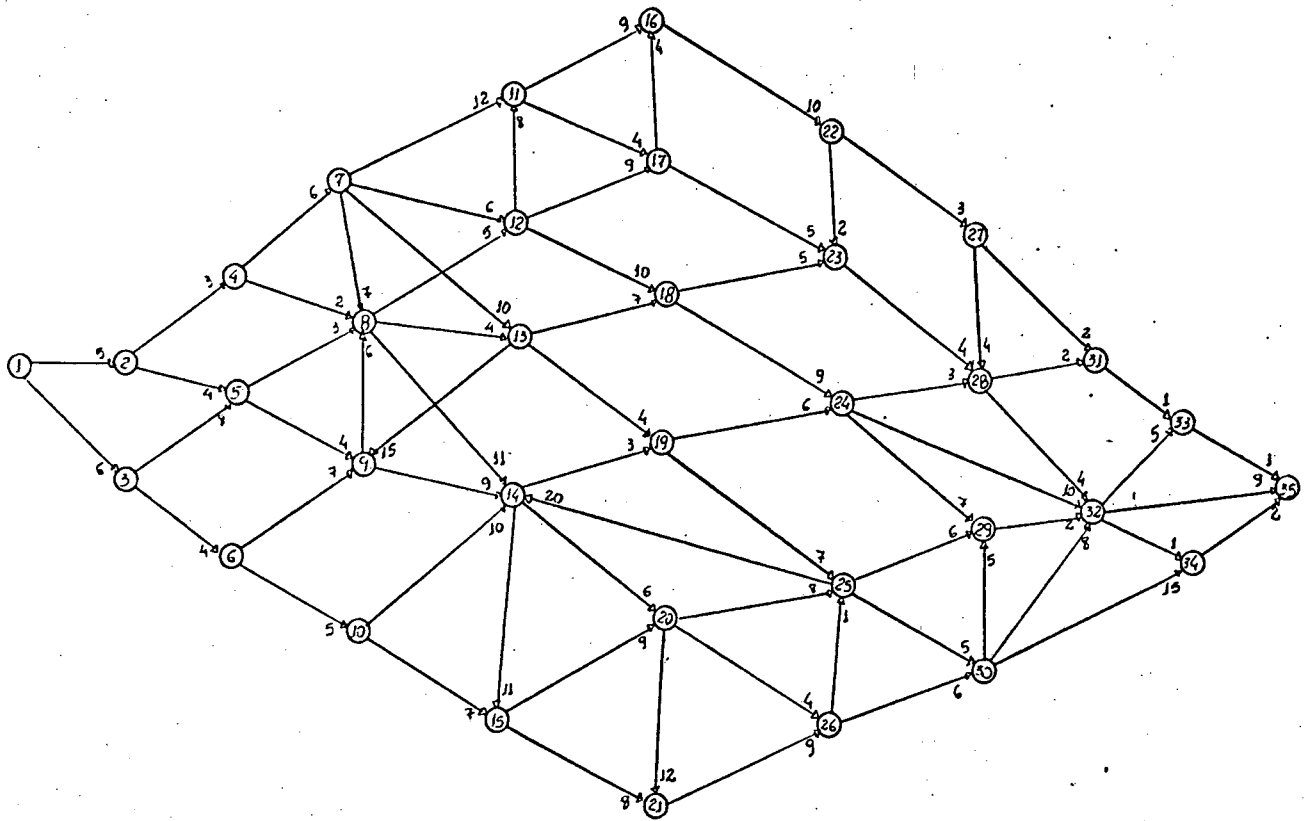
GRAFOS COM 35 VÉRTICES

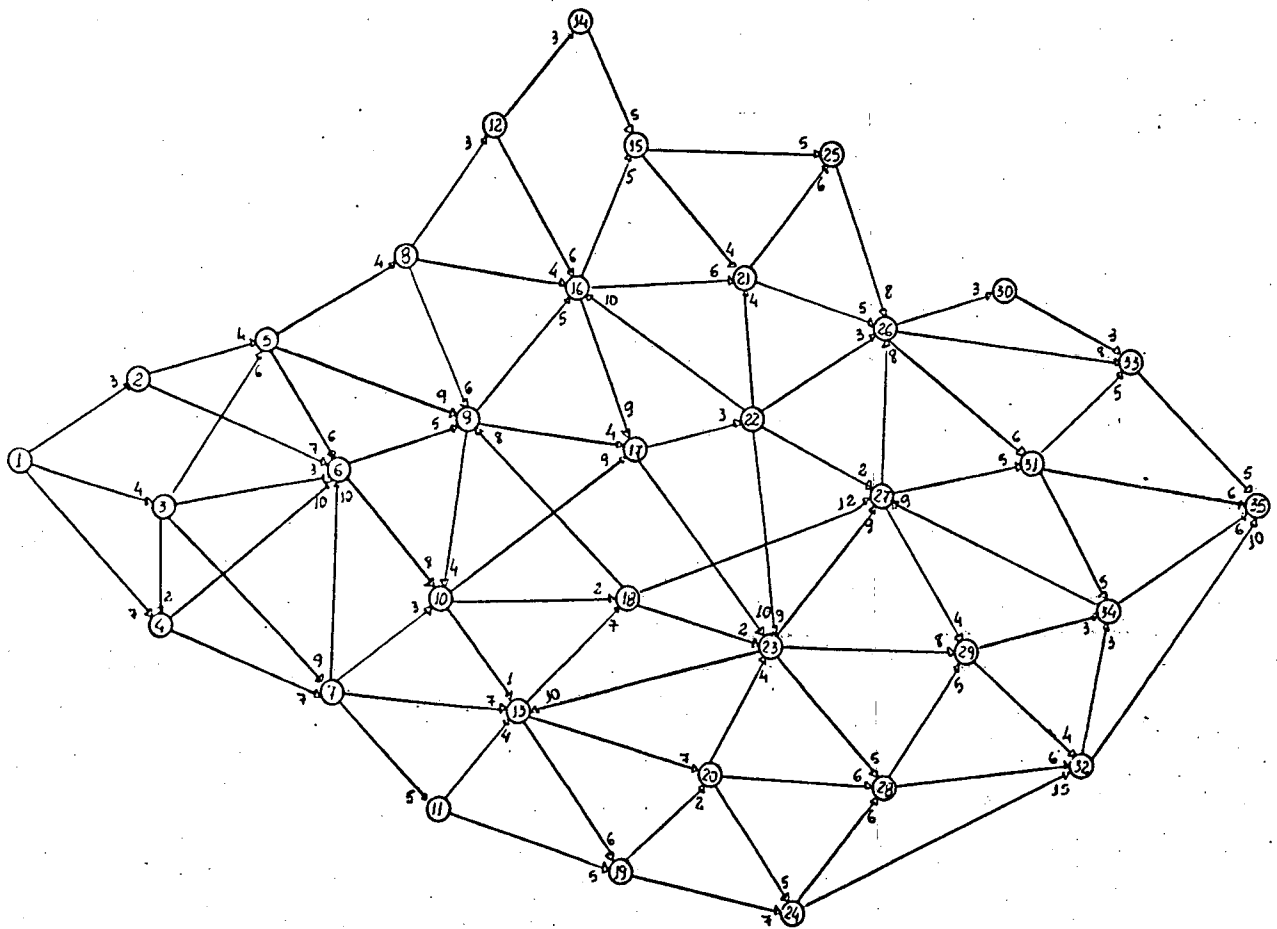
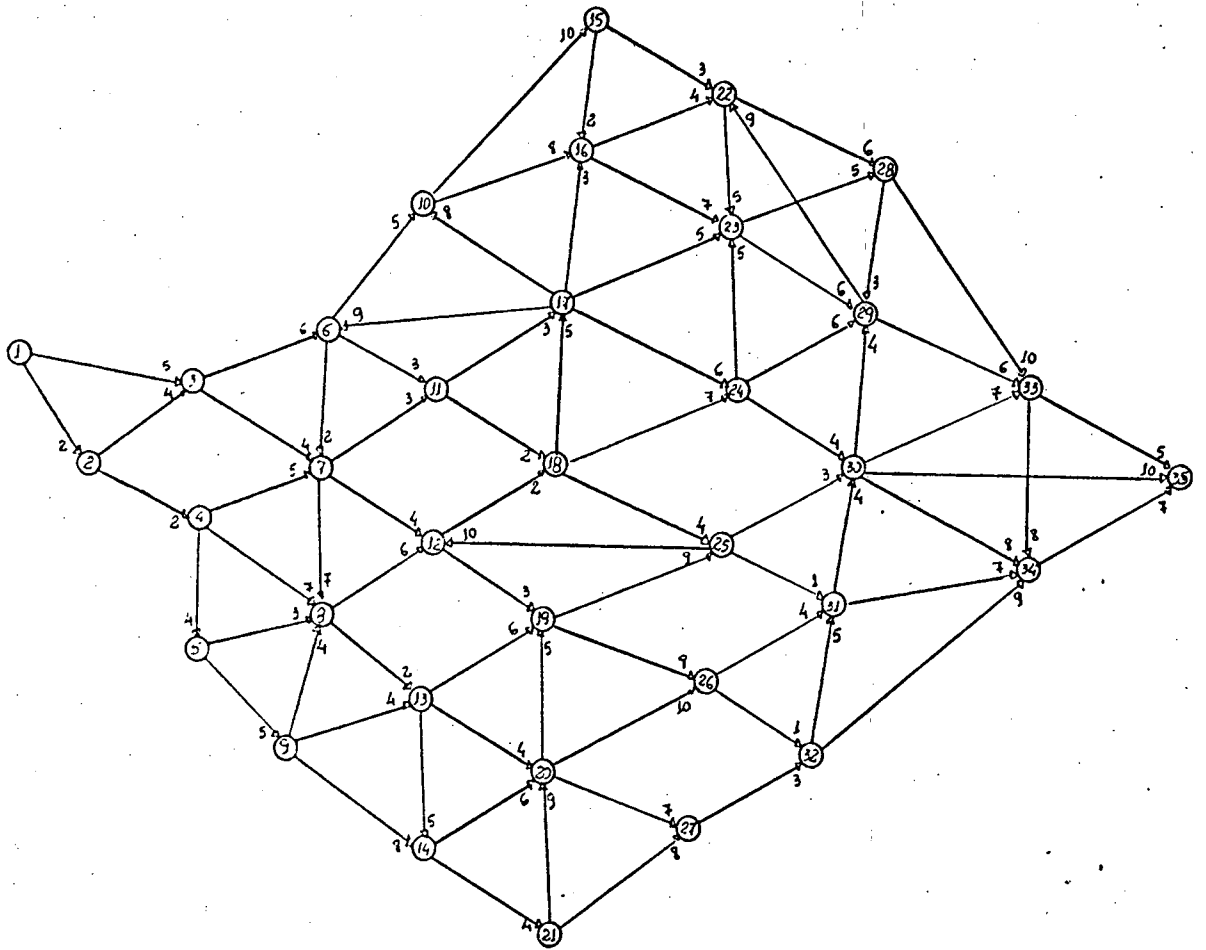


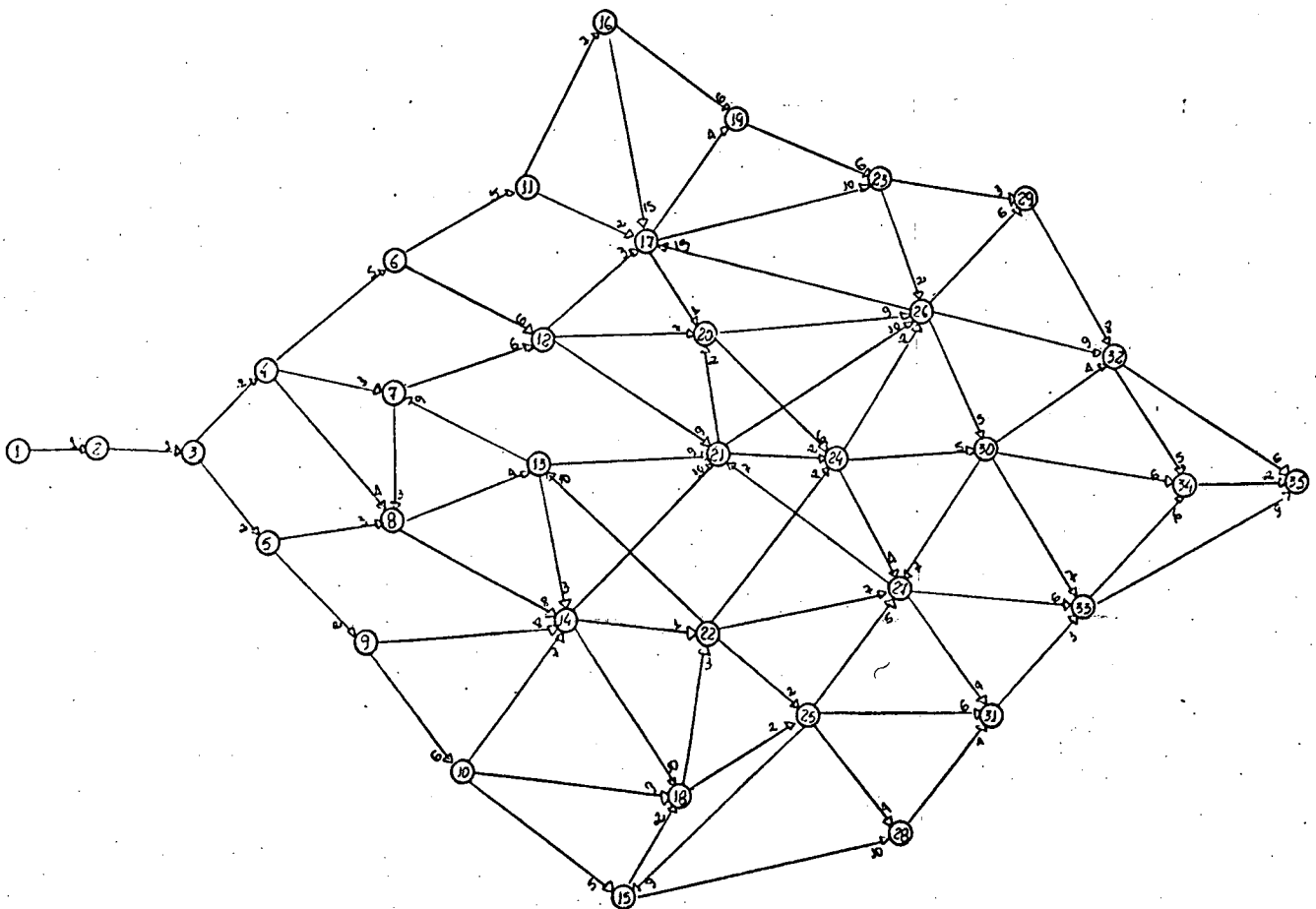
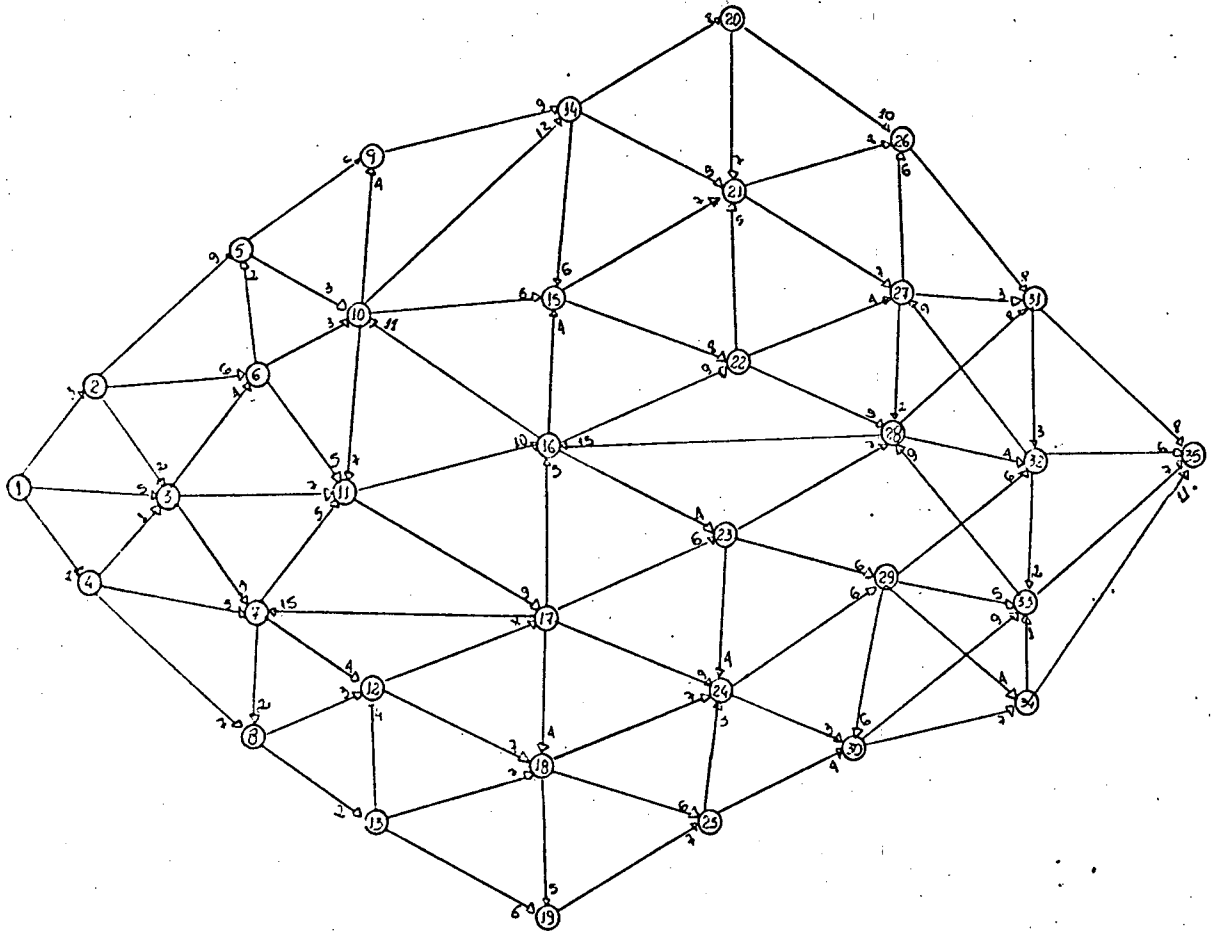


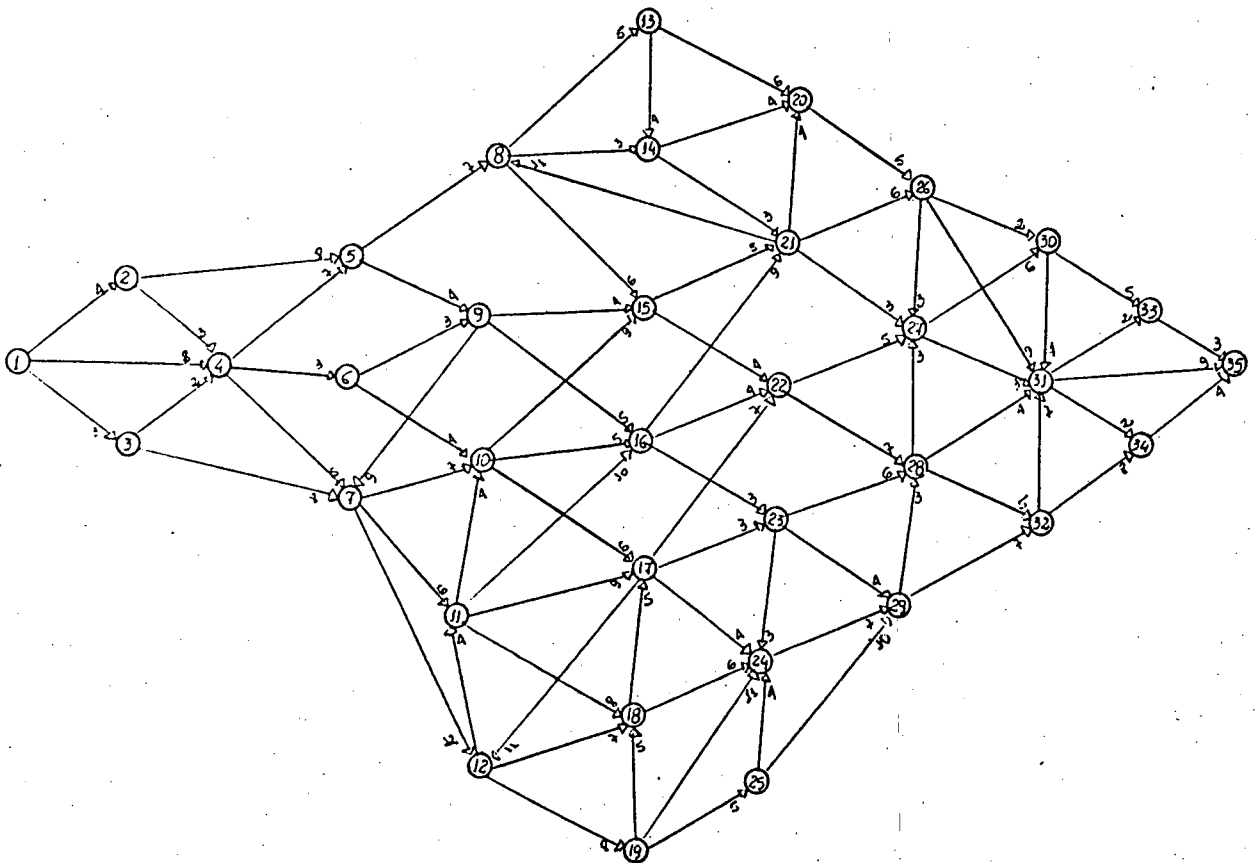
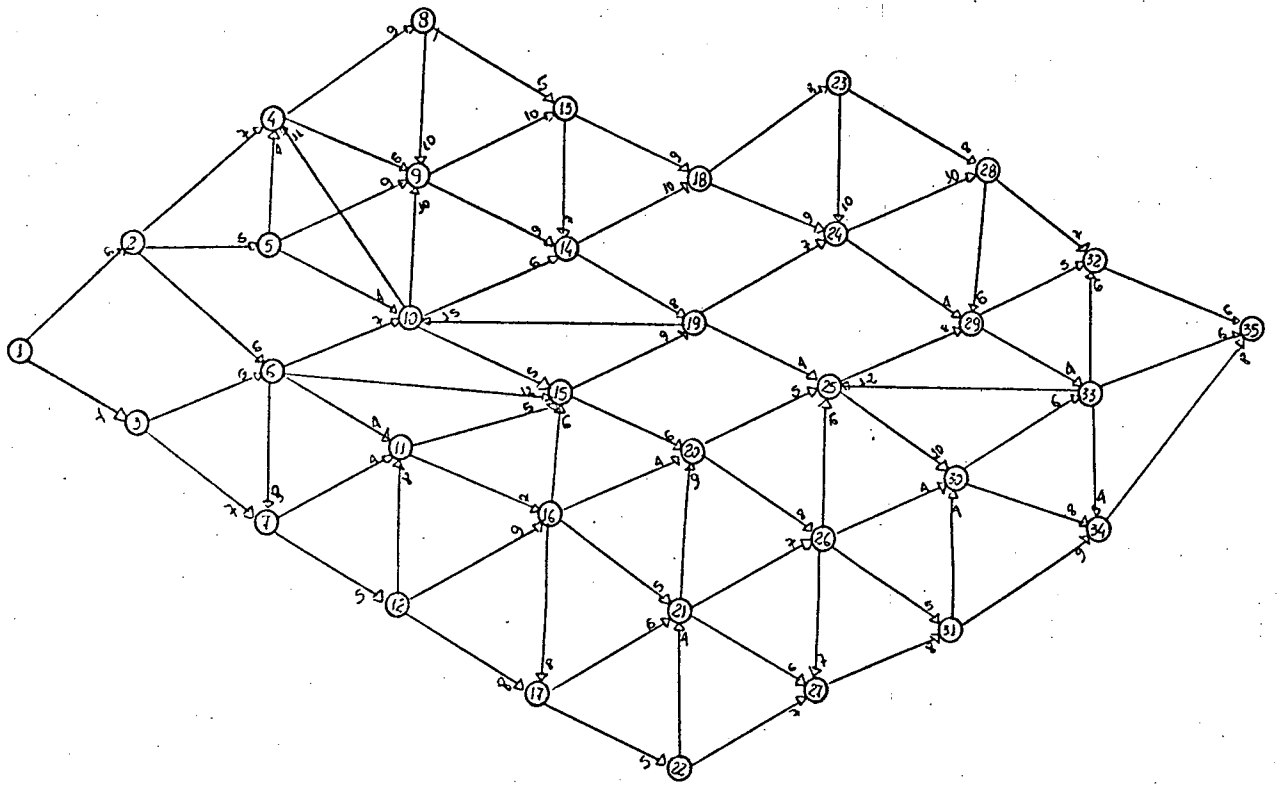


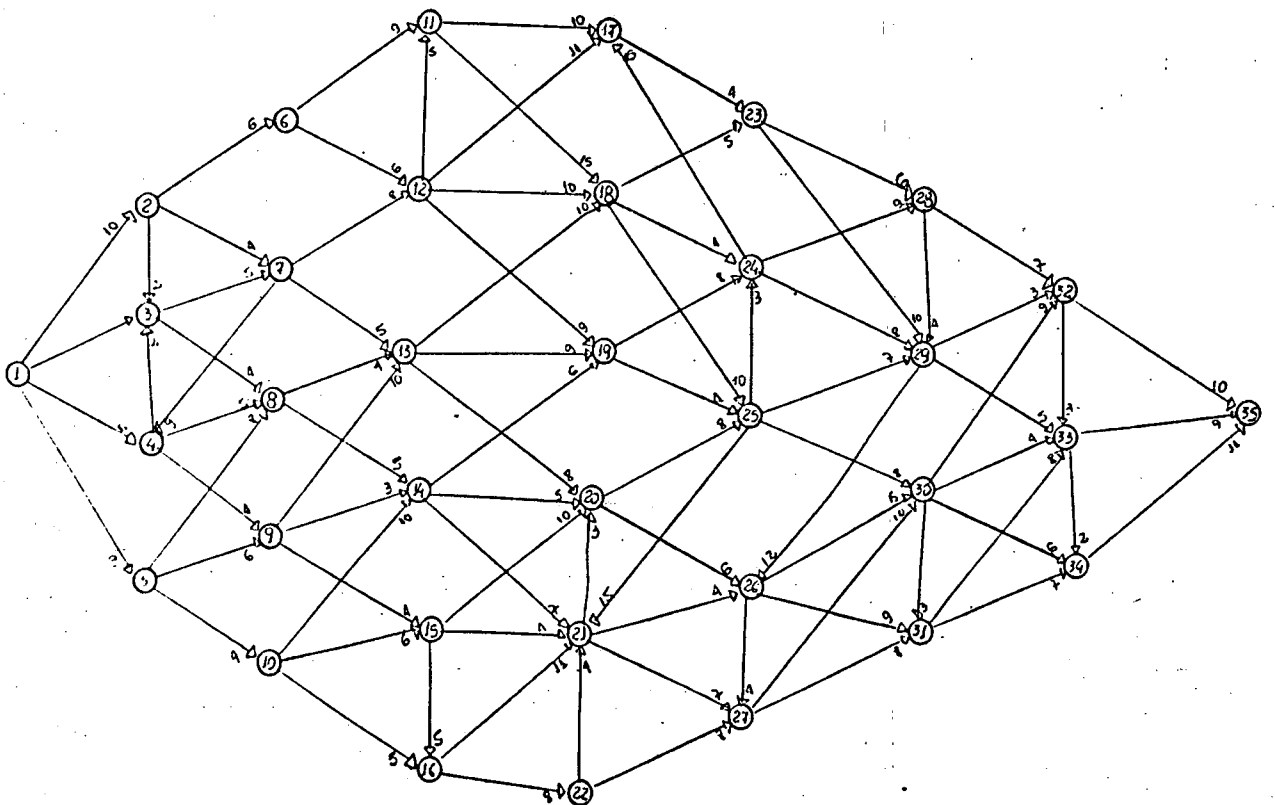
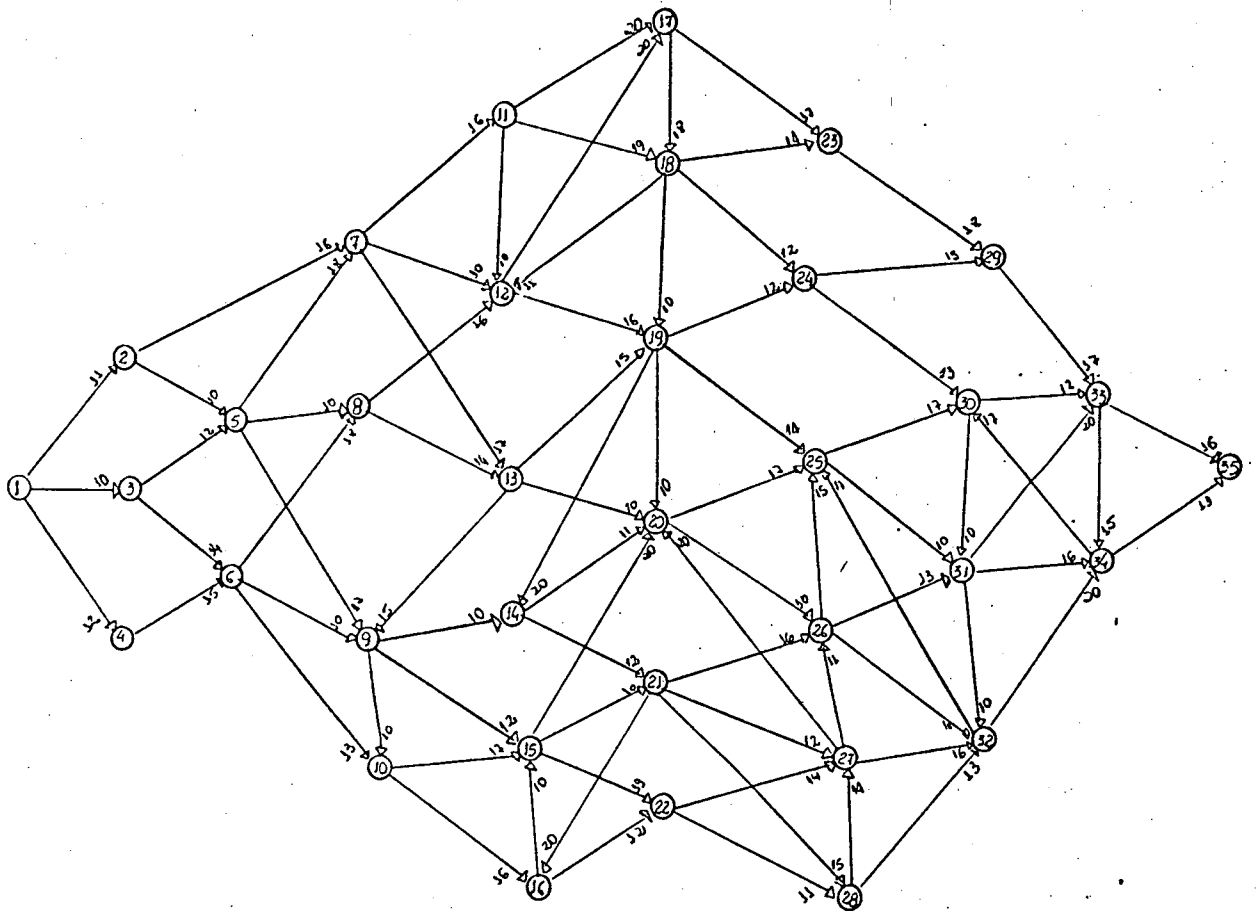


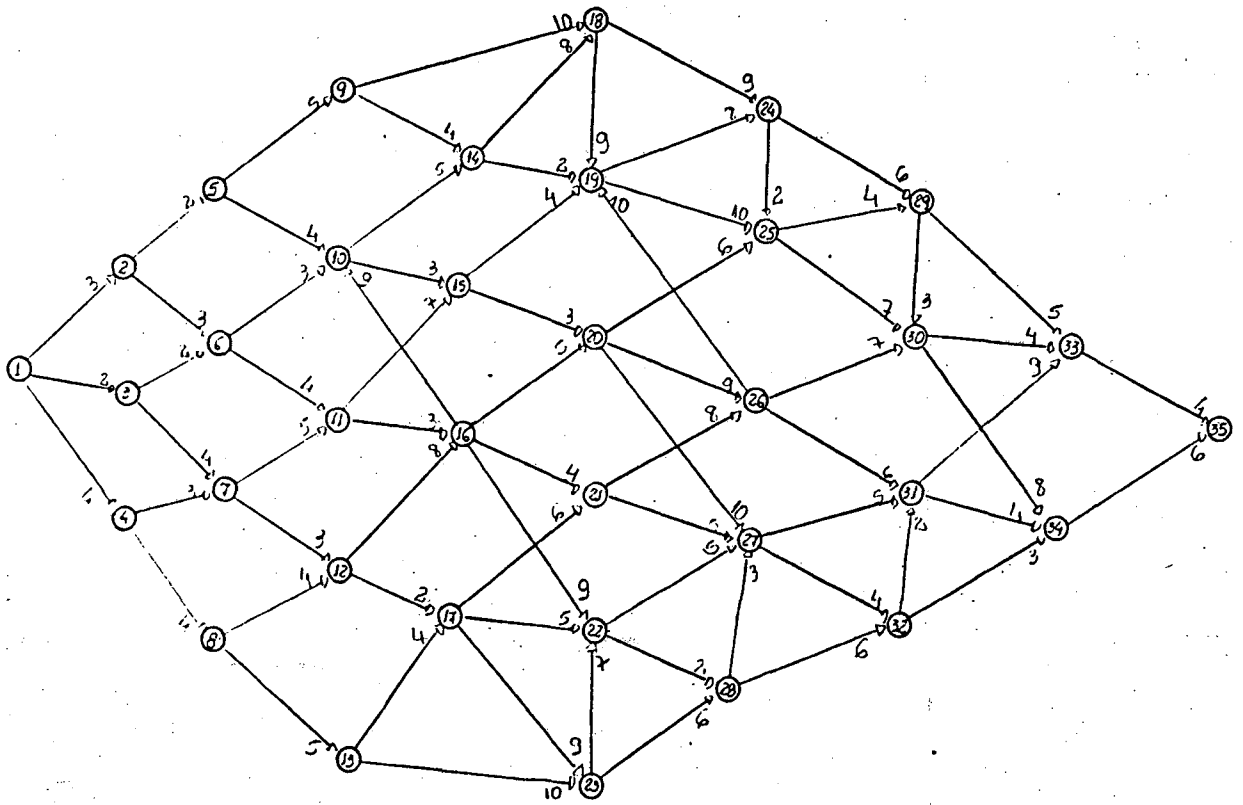
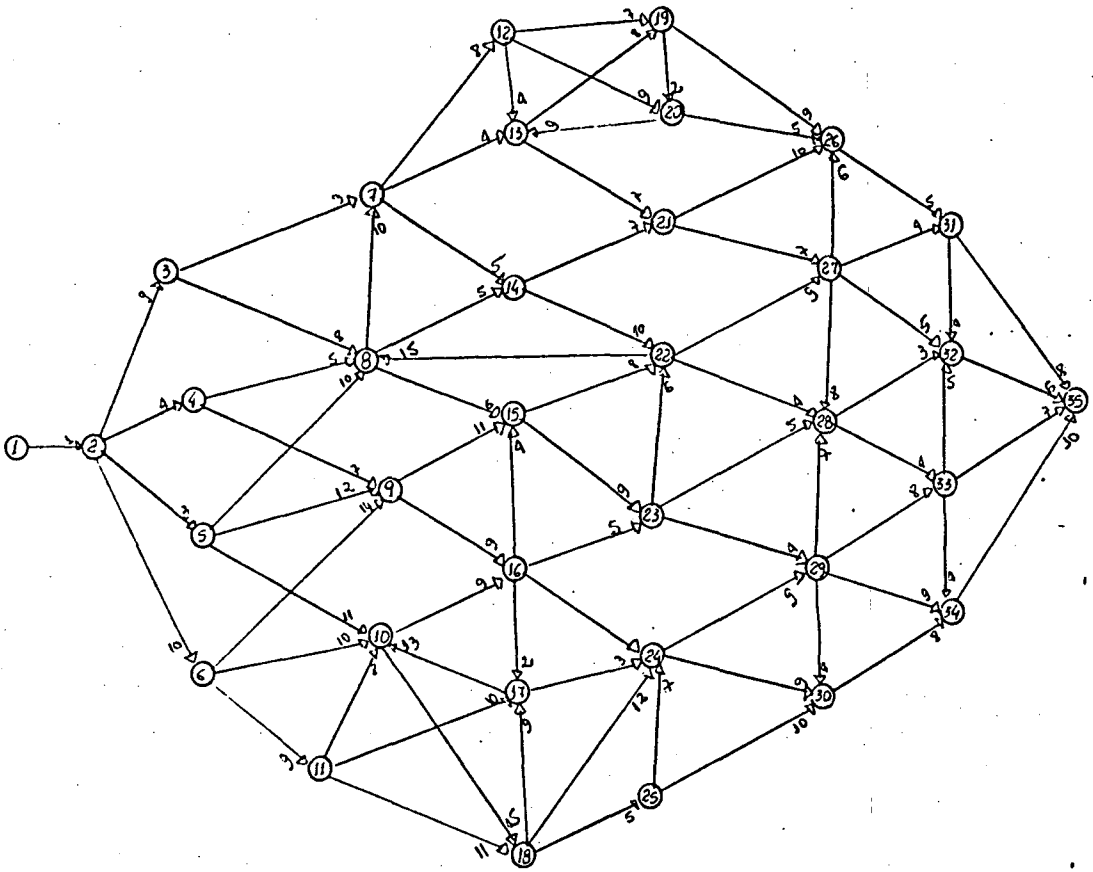






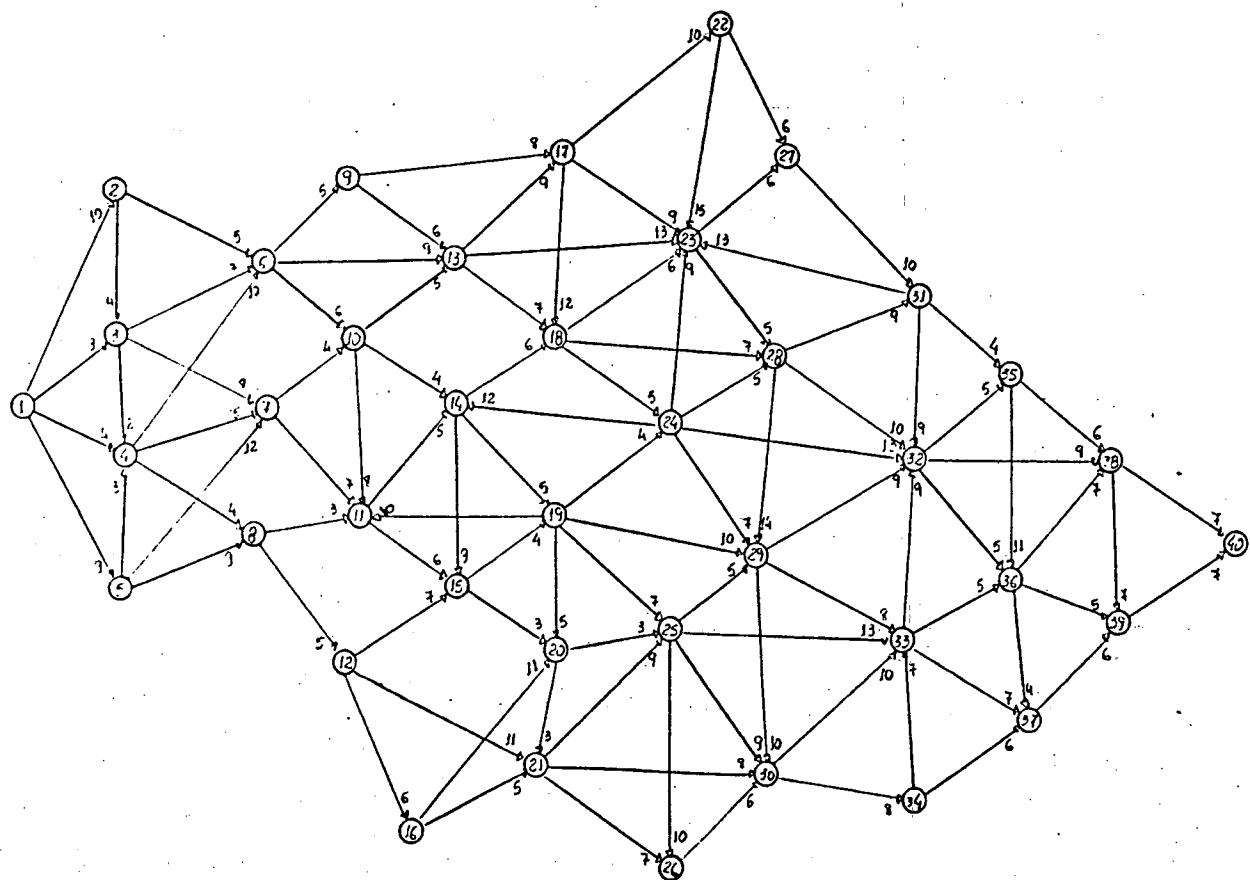
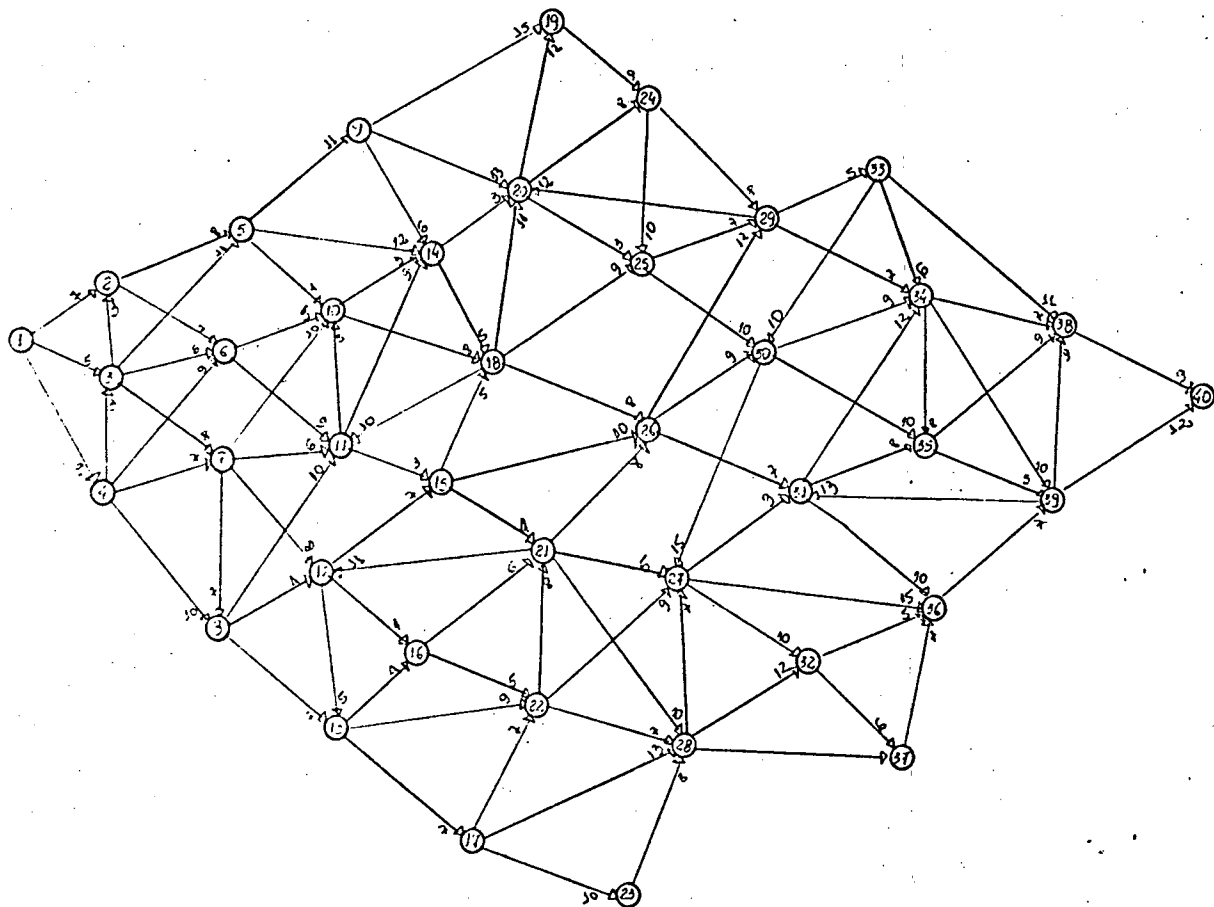


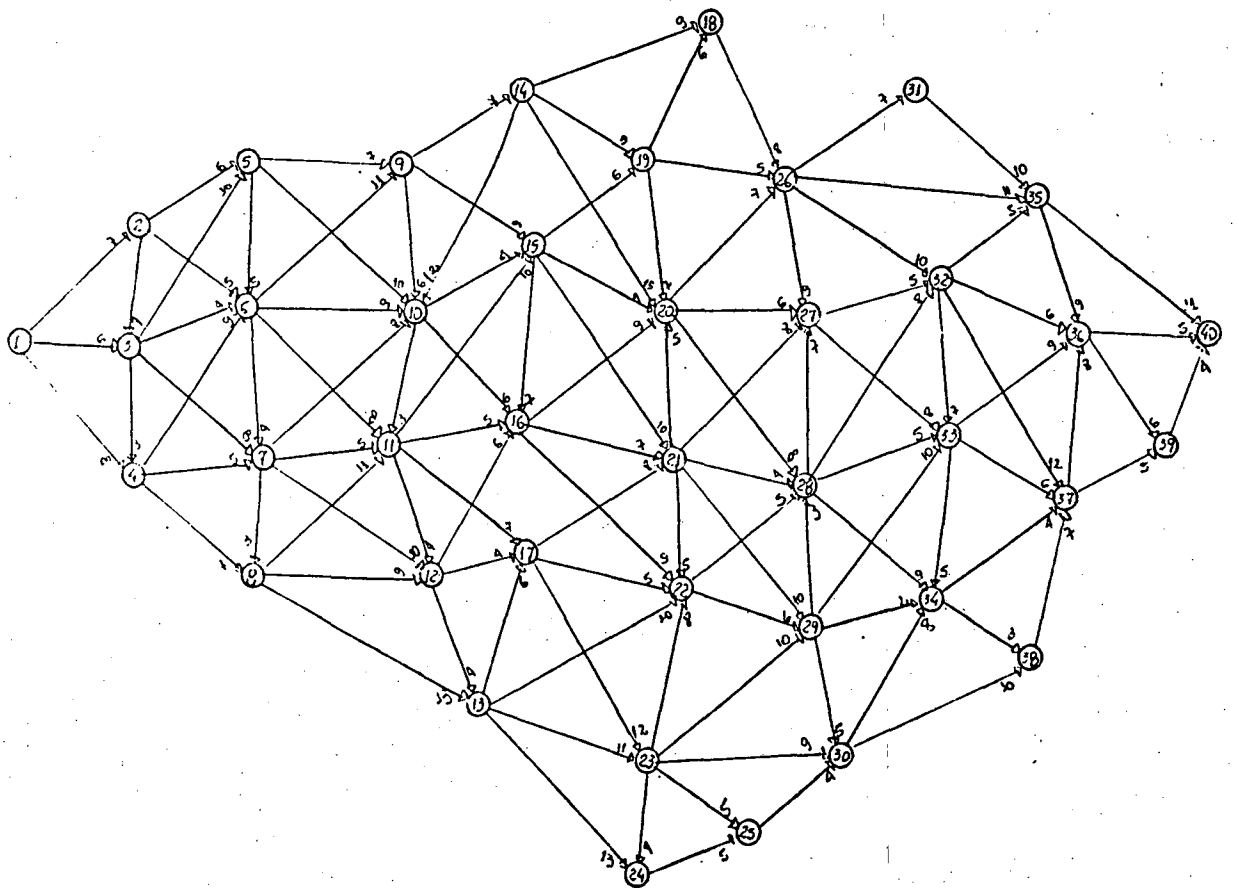
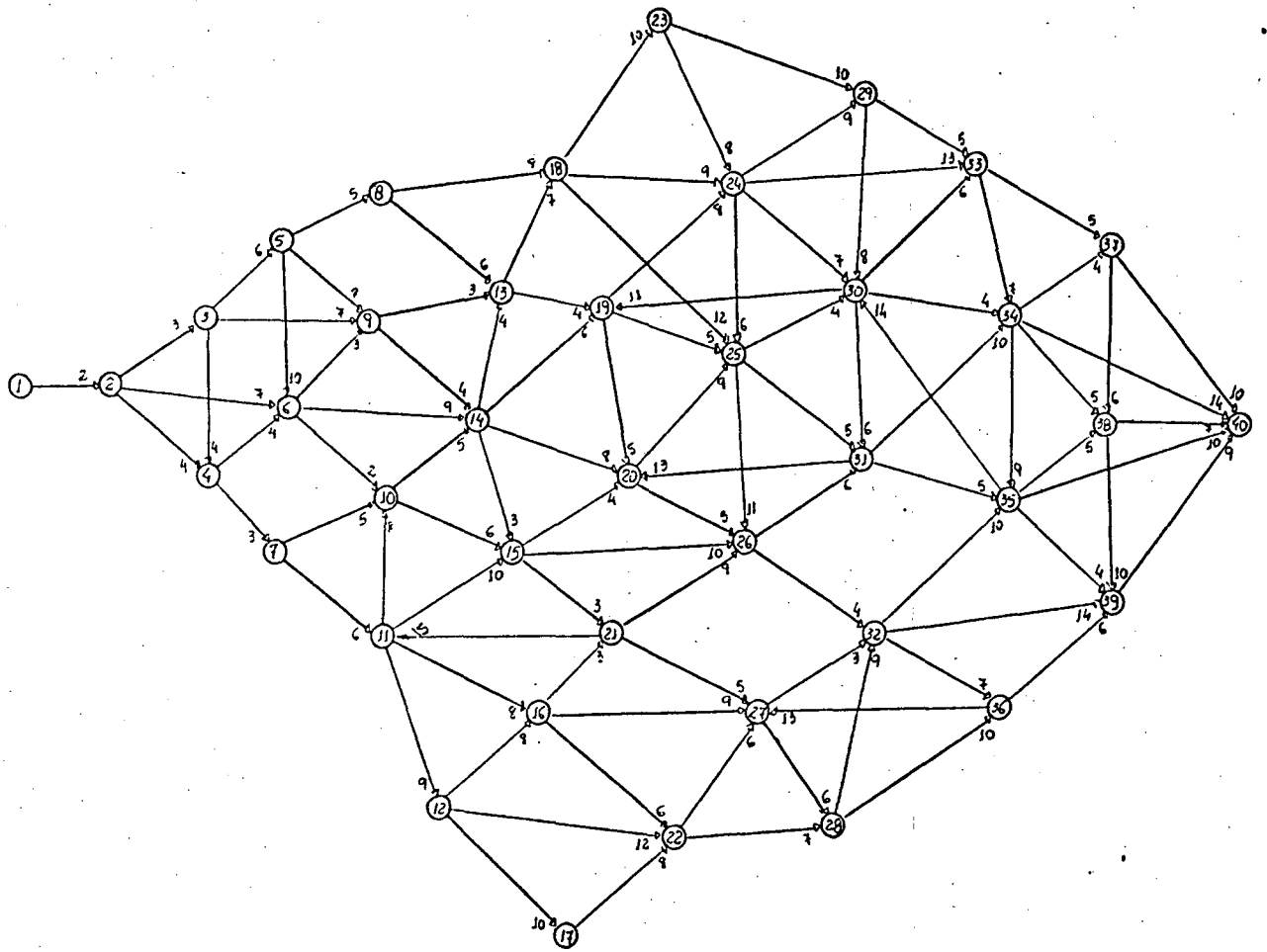


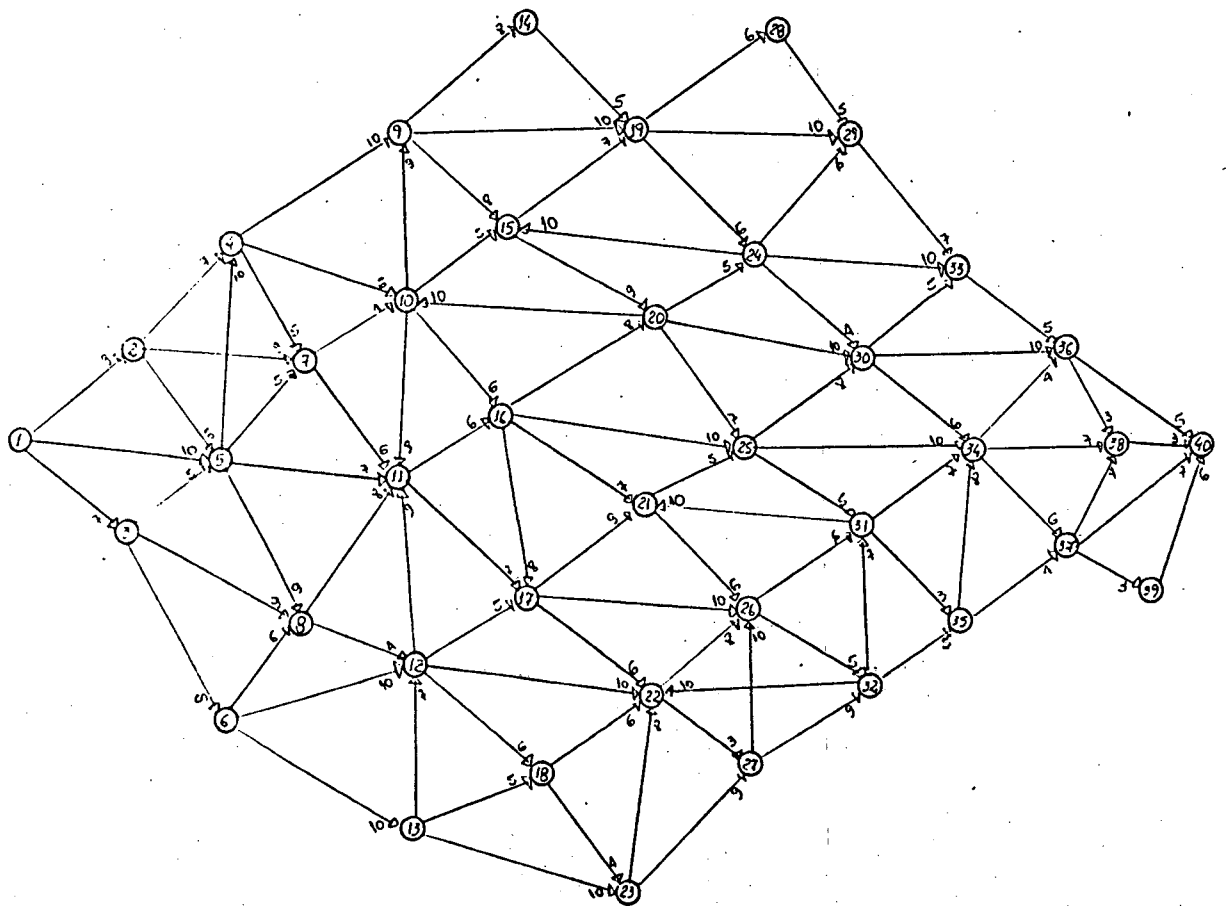
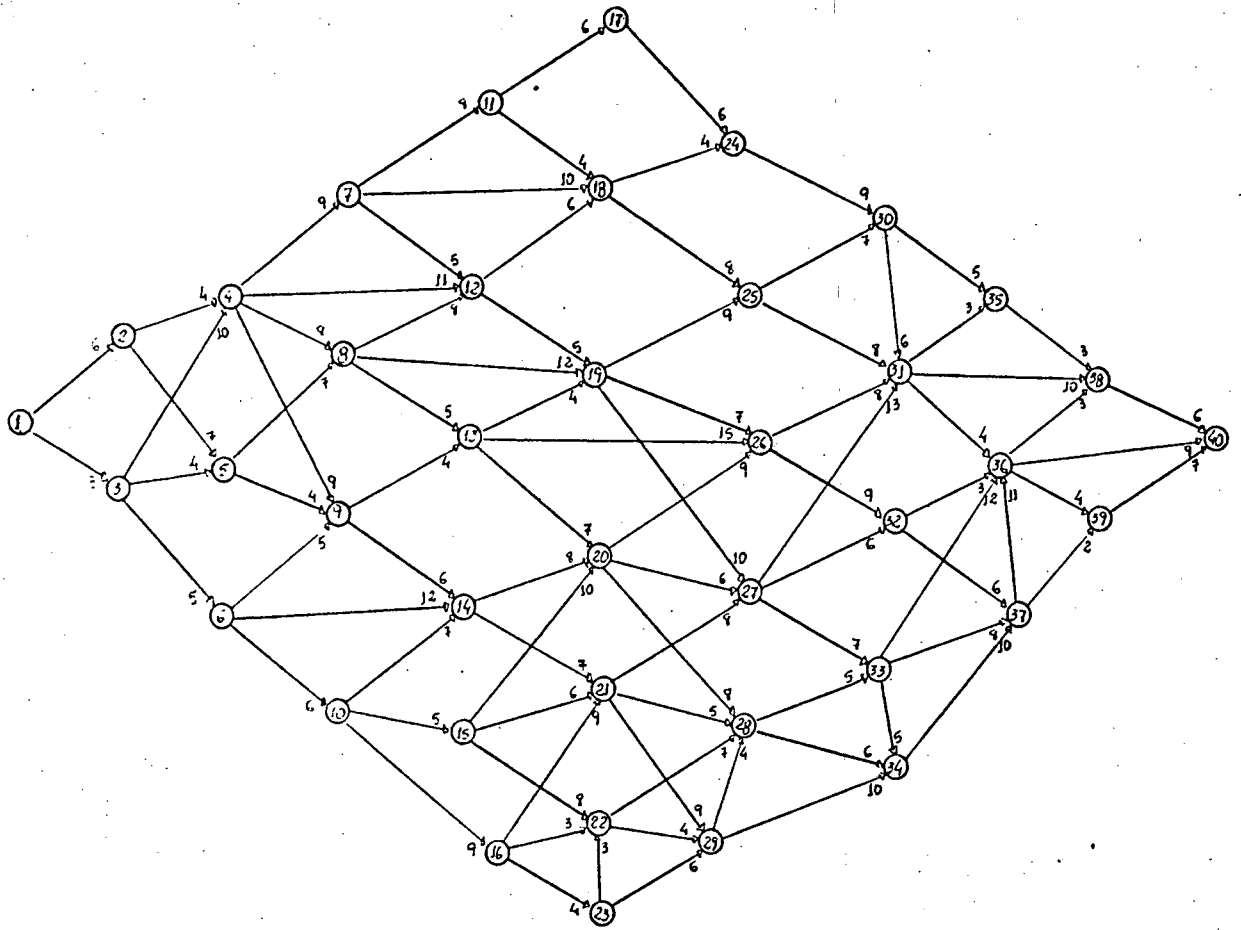


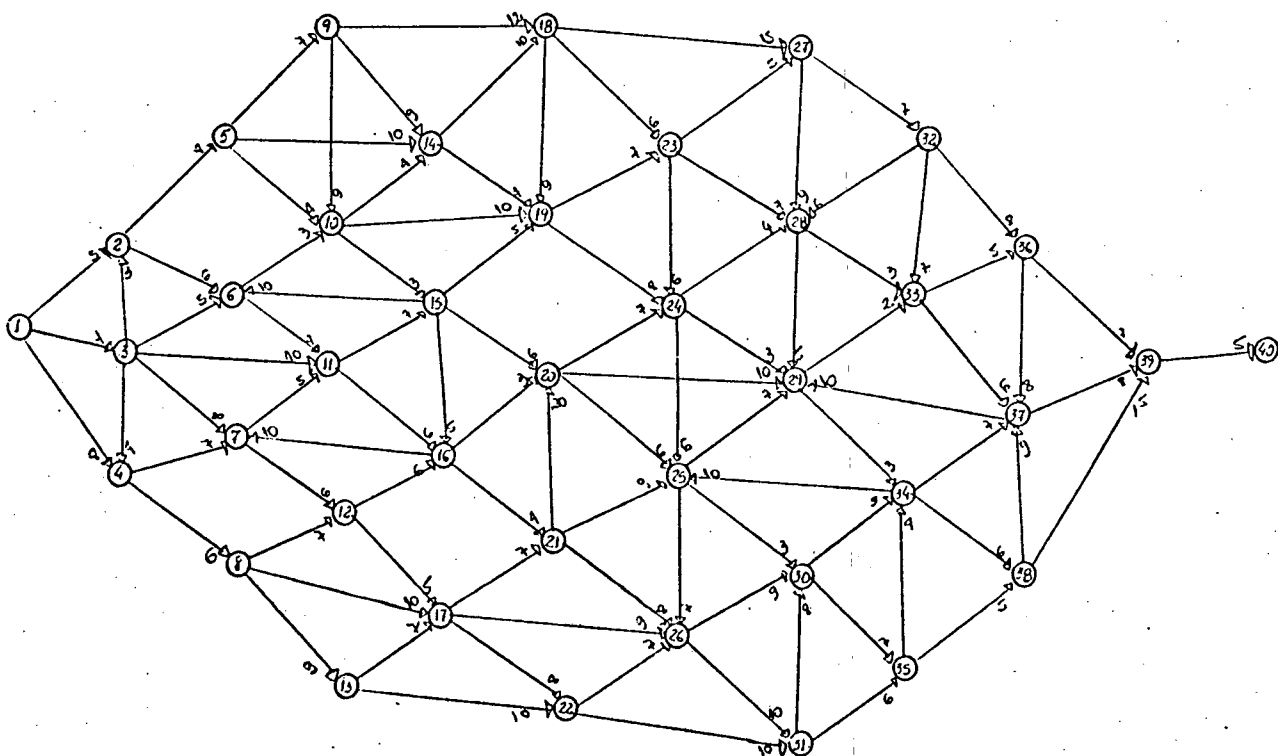
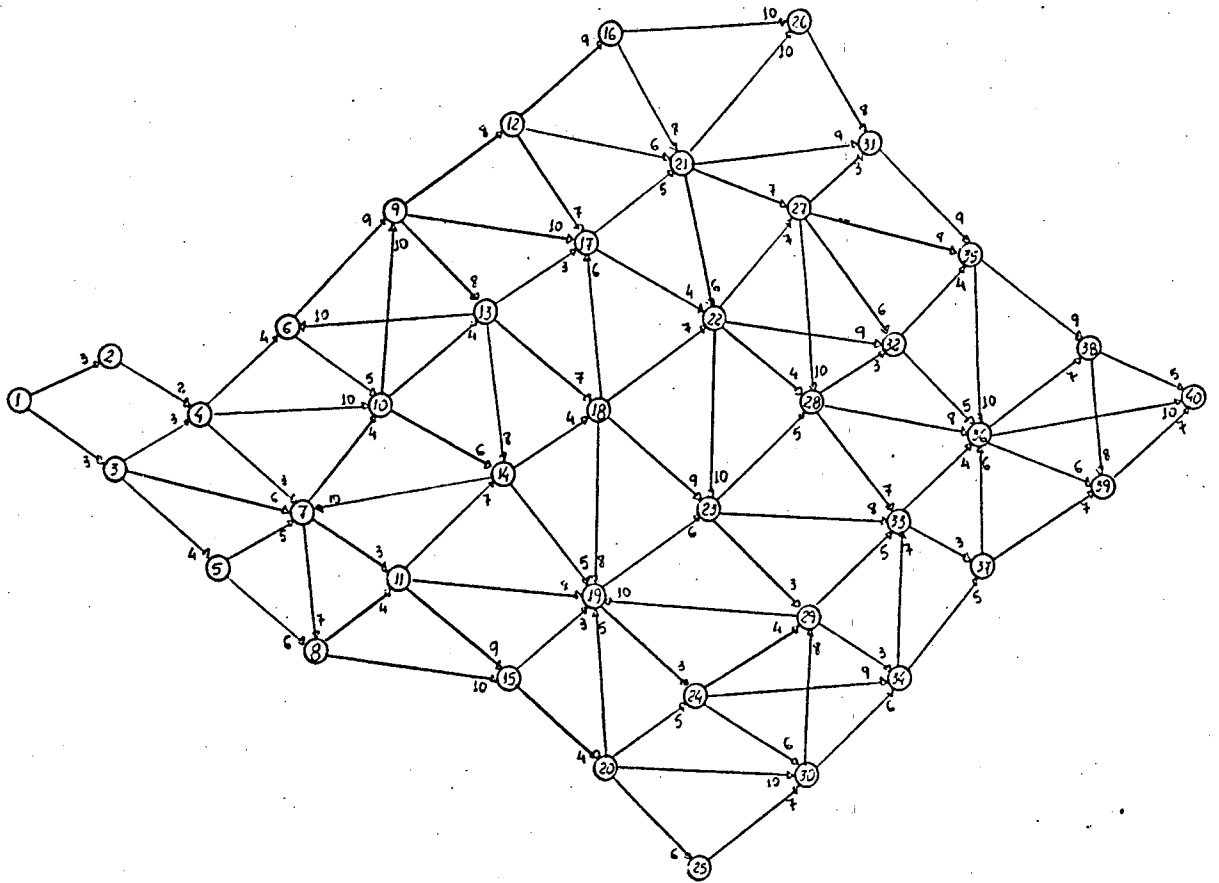
F I G U R A 12

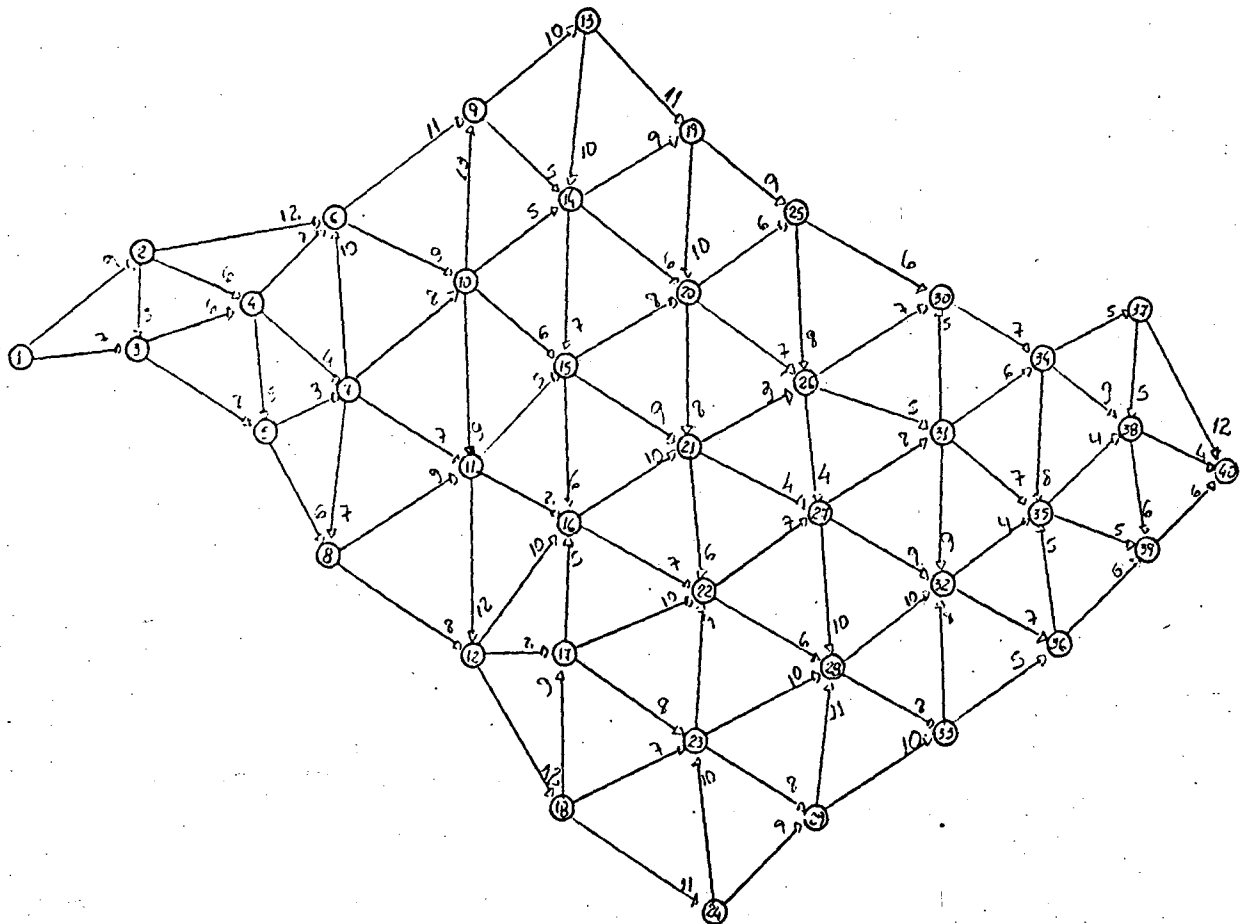
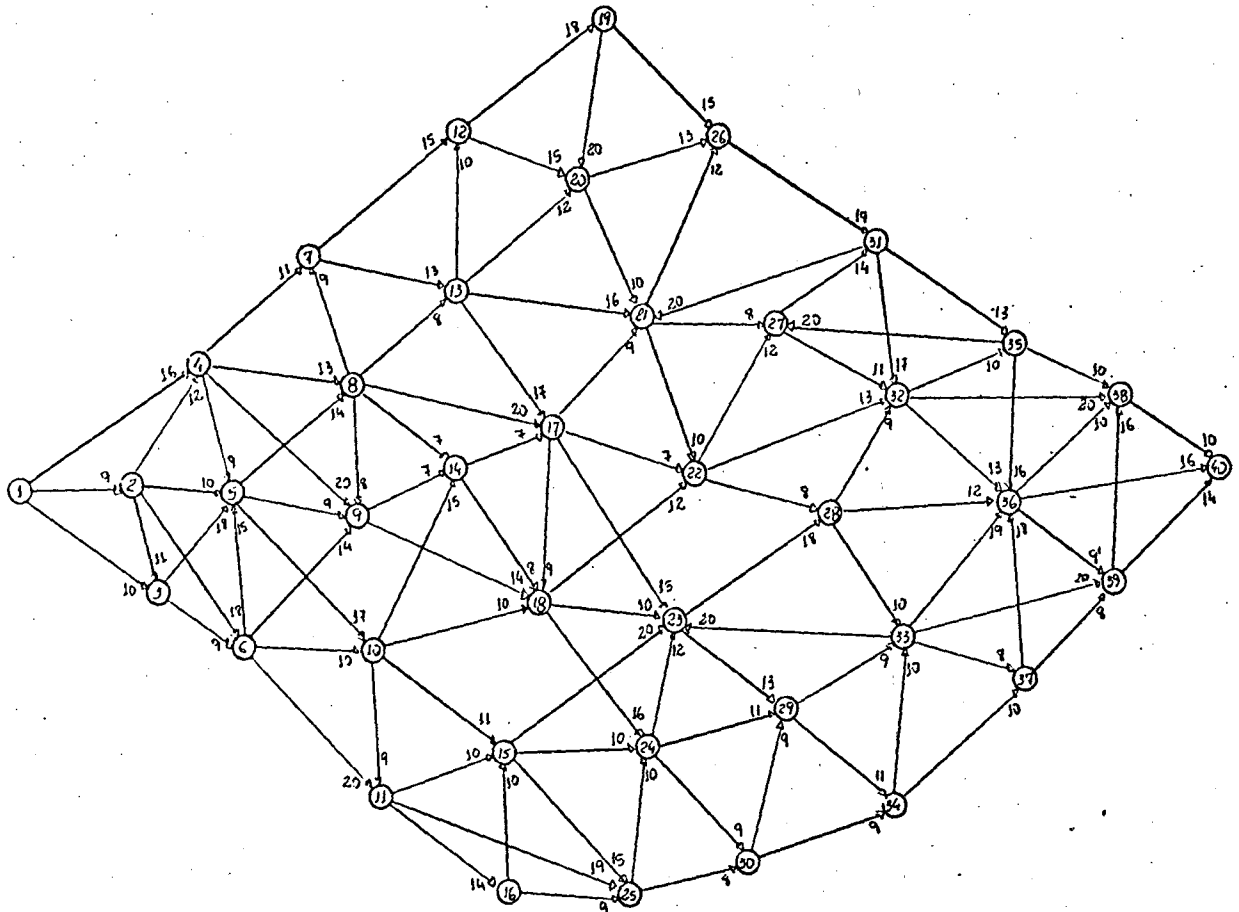
GRAFOS COM 40 VÉRTICES

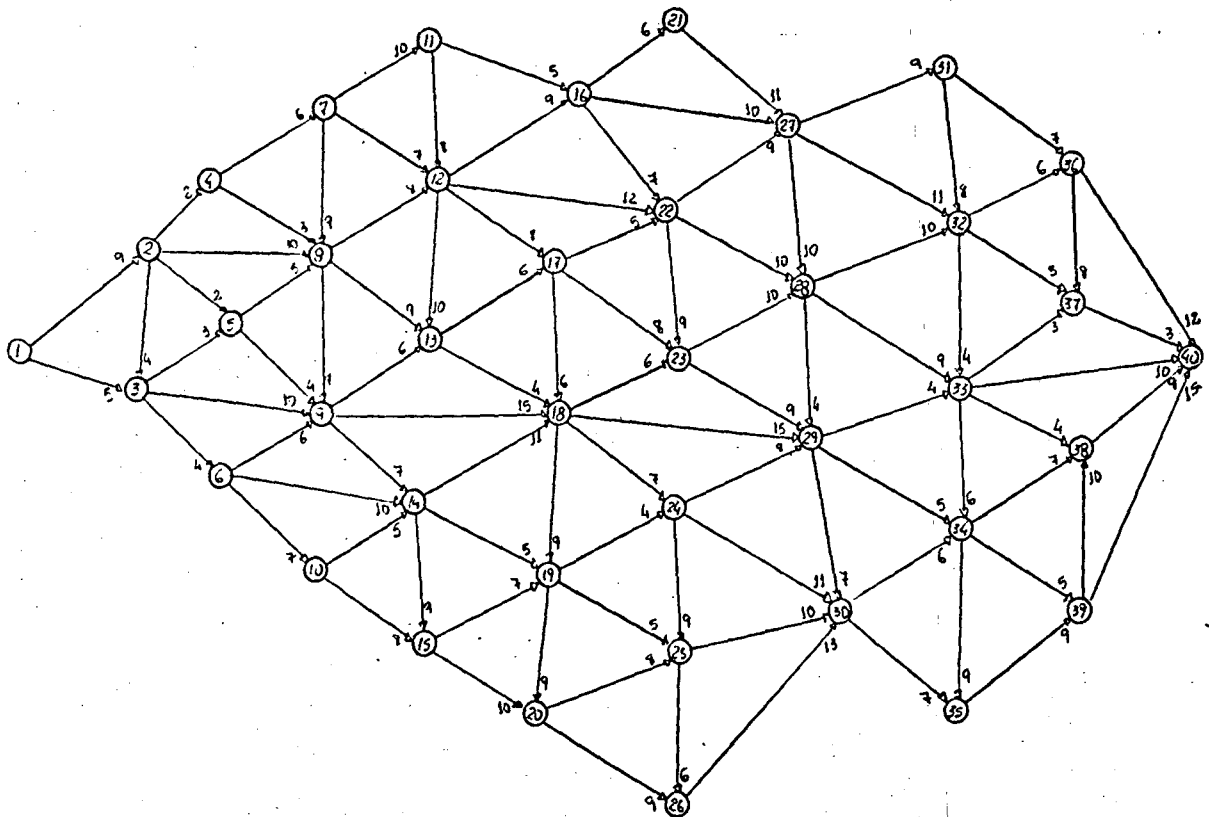
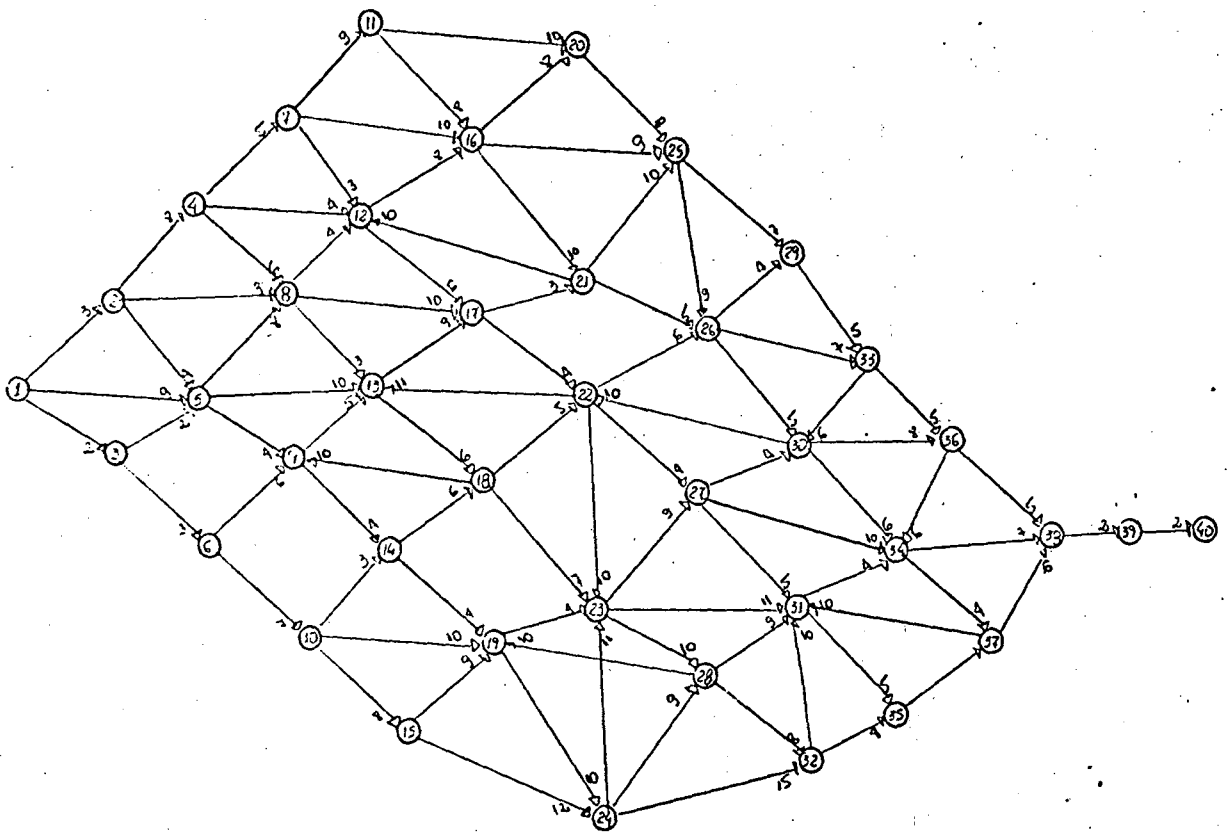


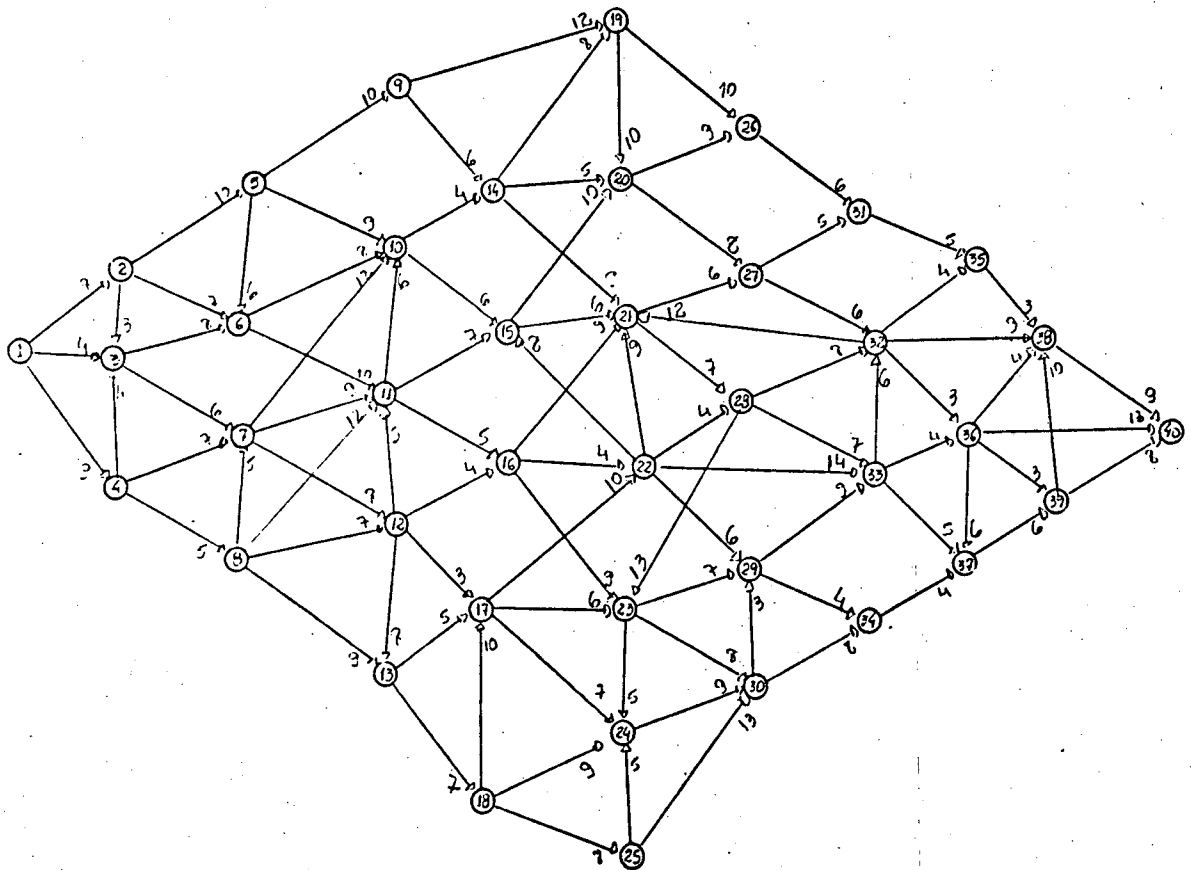
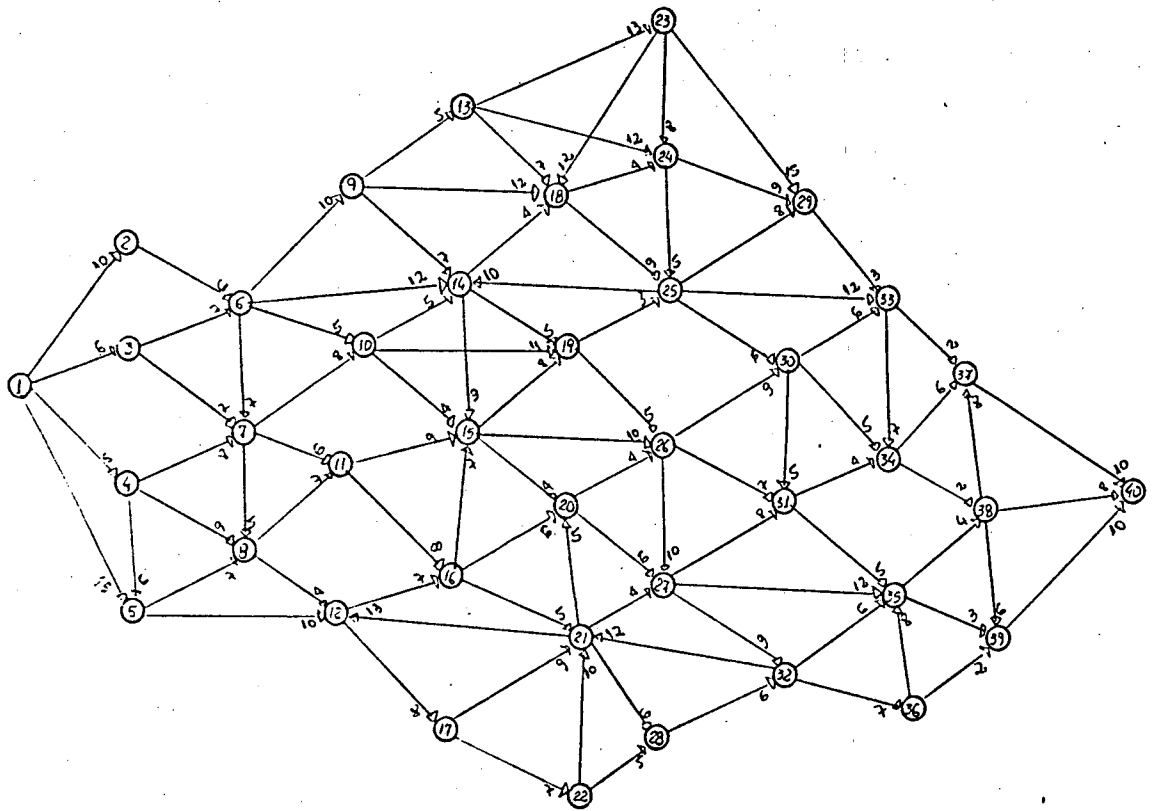


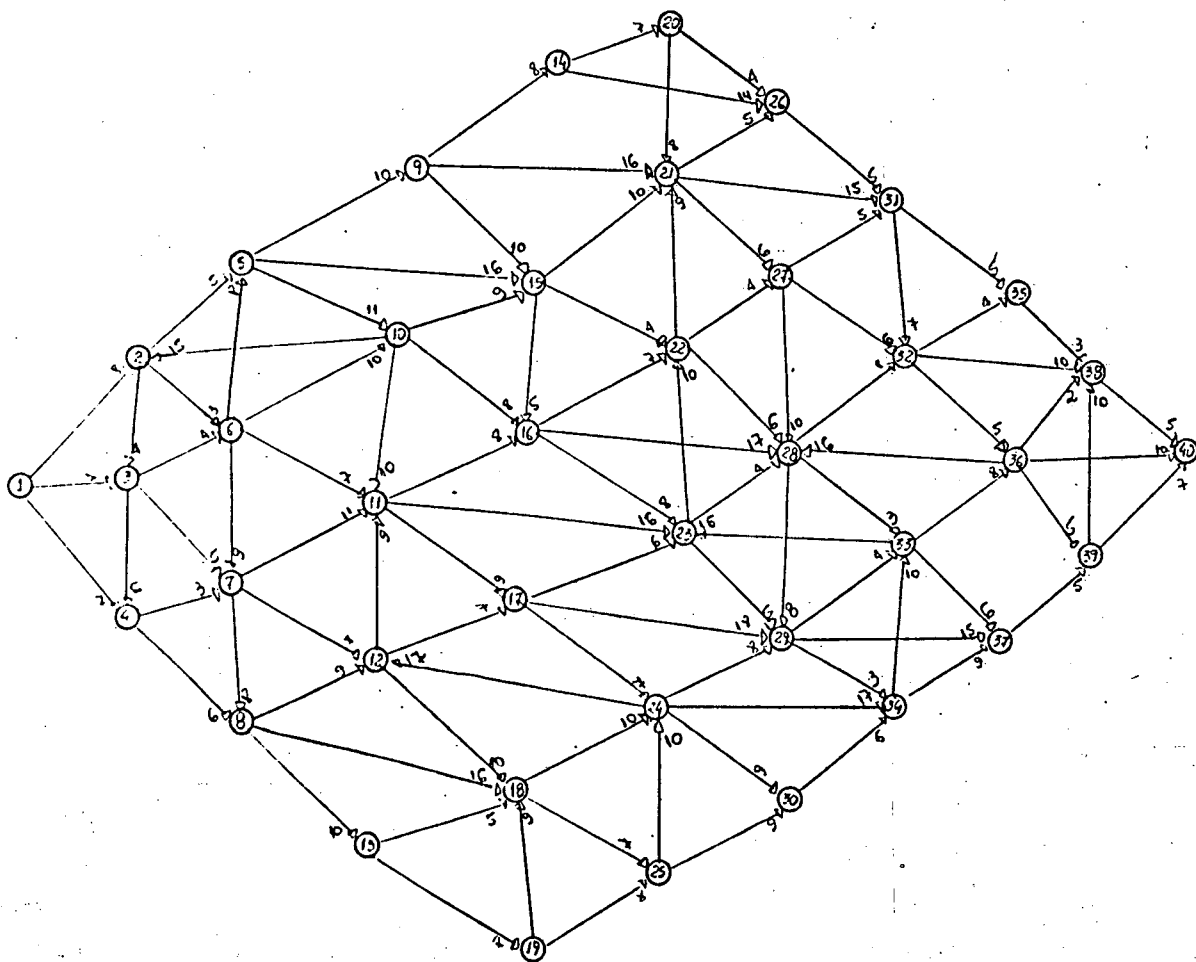
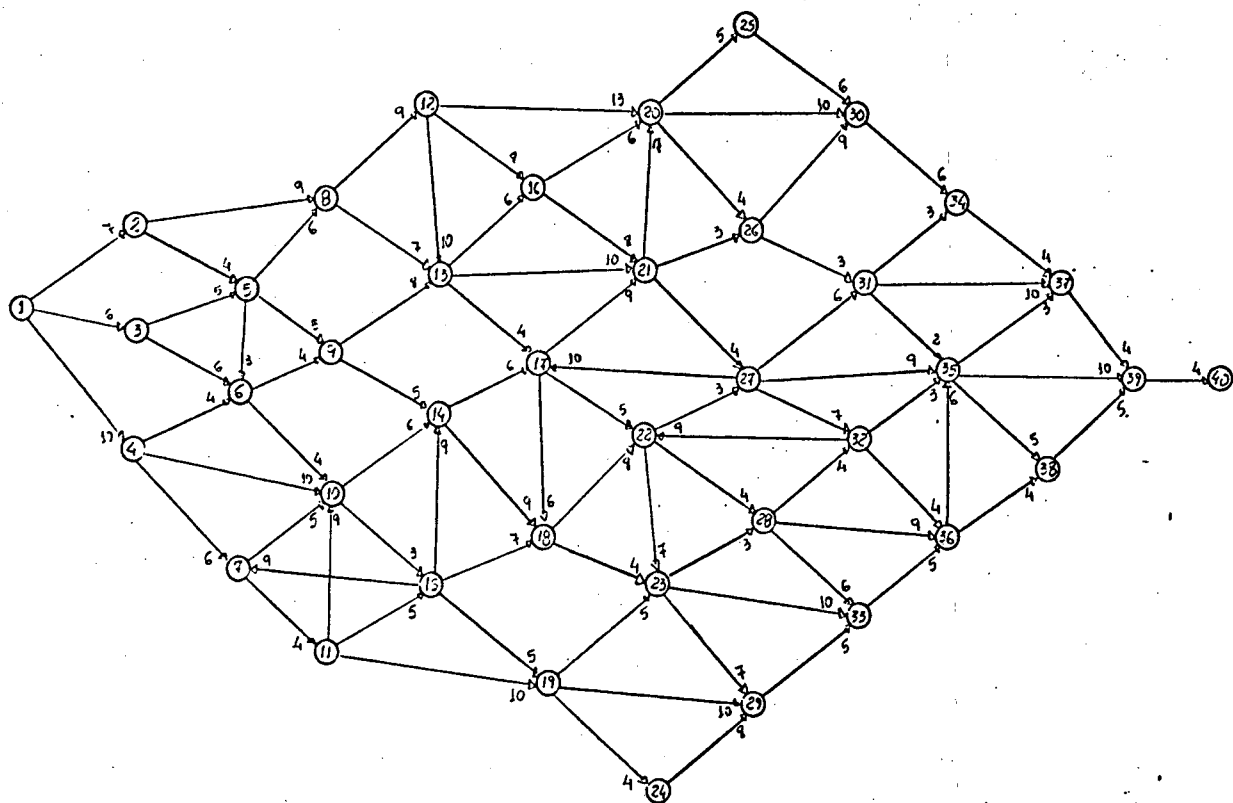


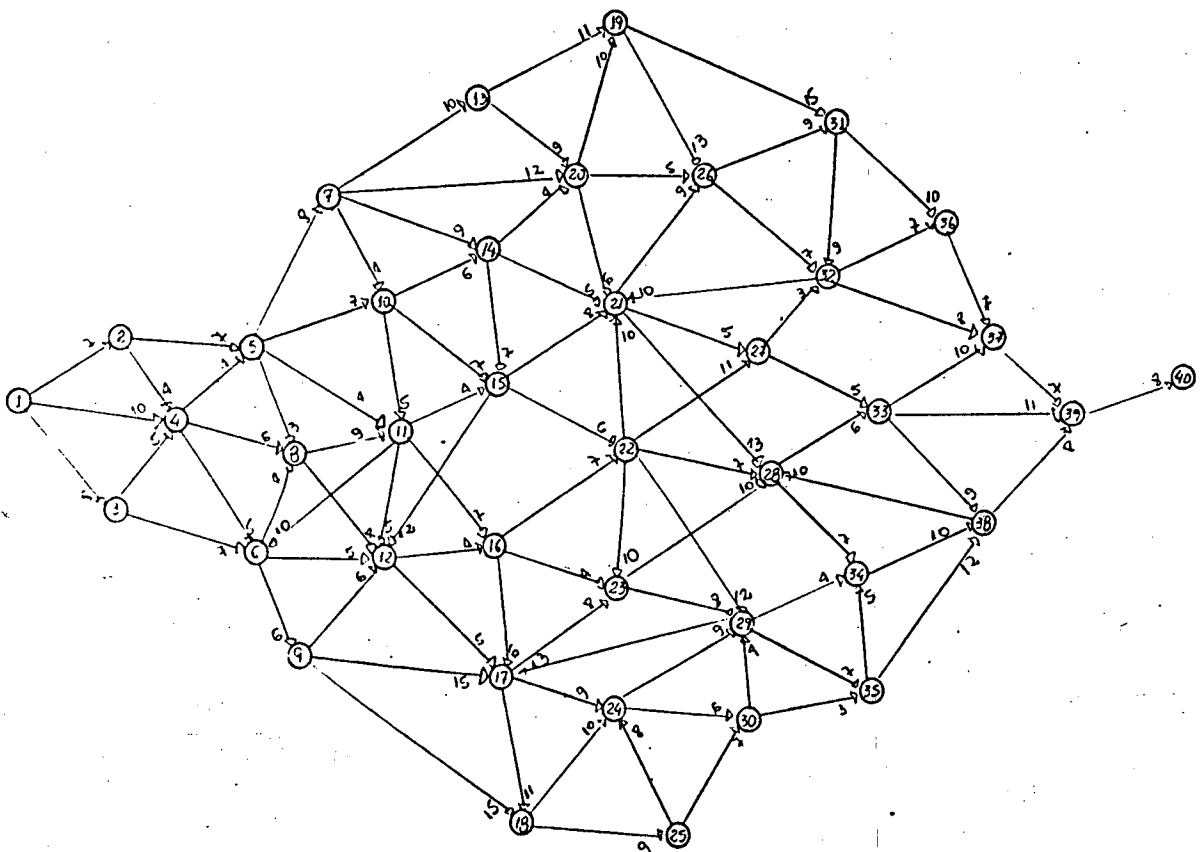
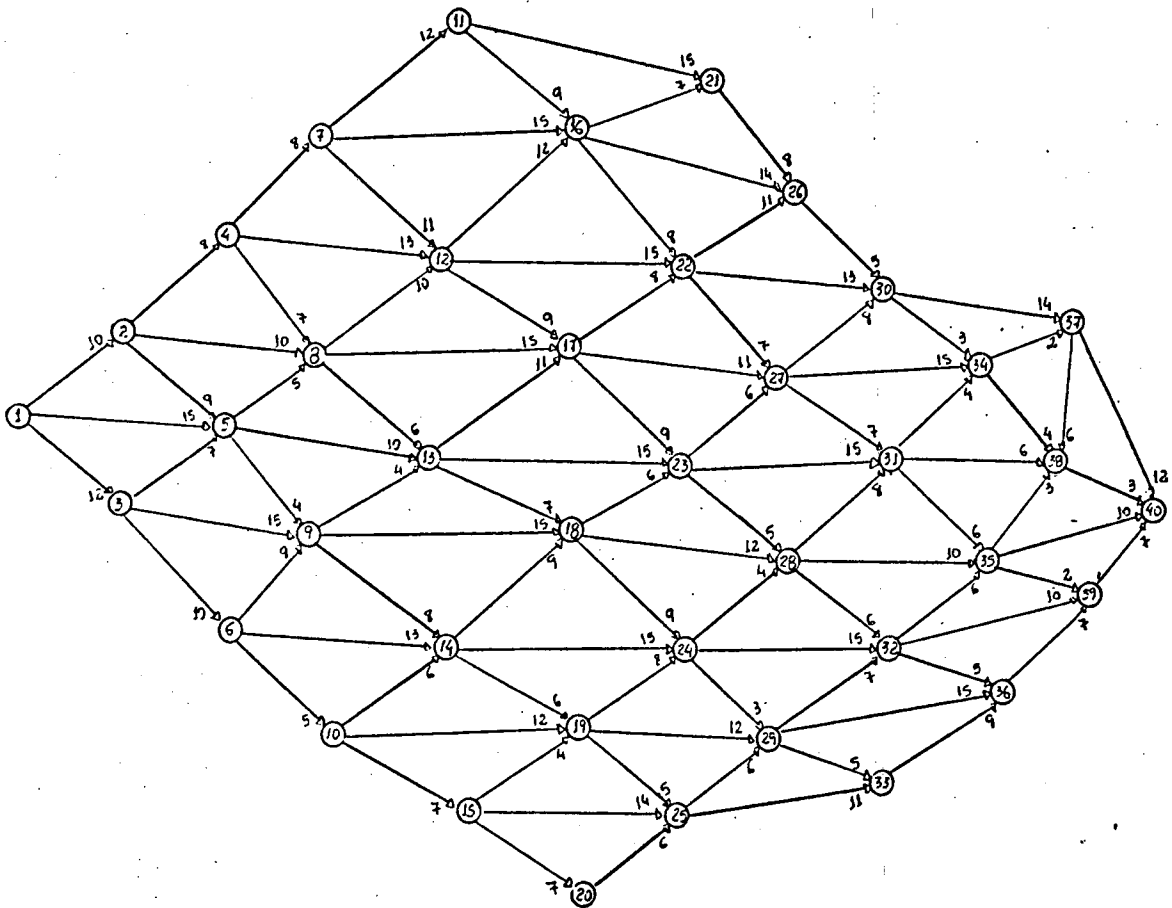


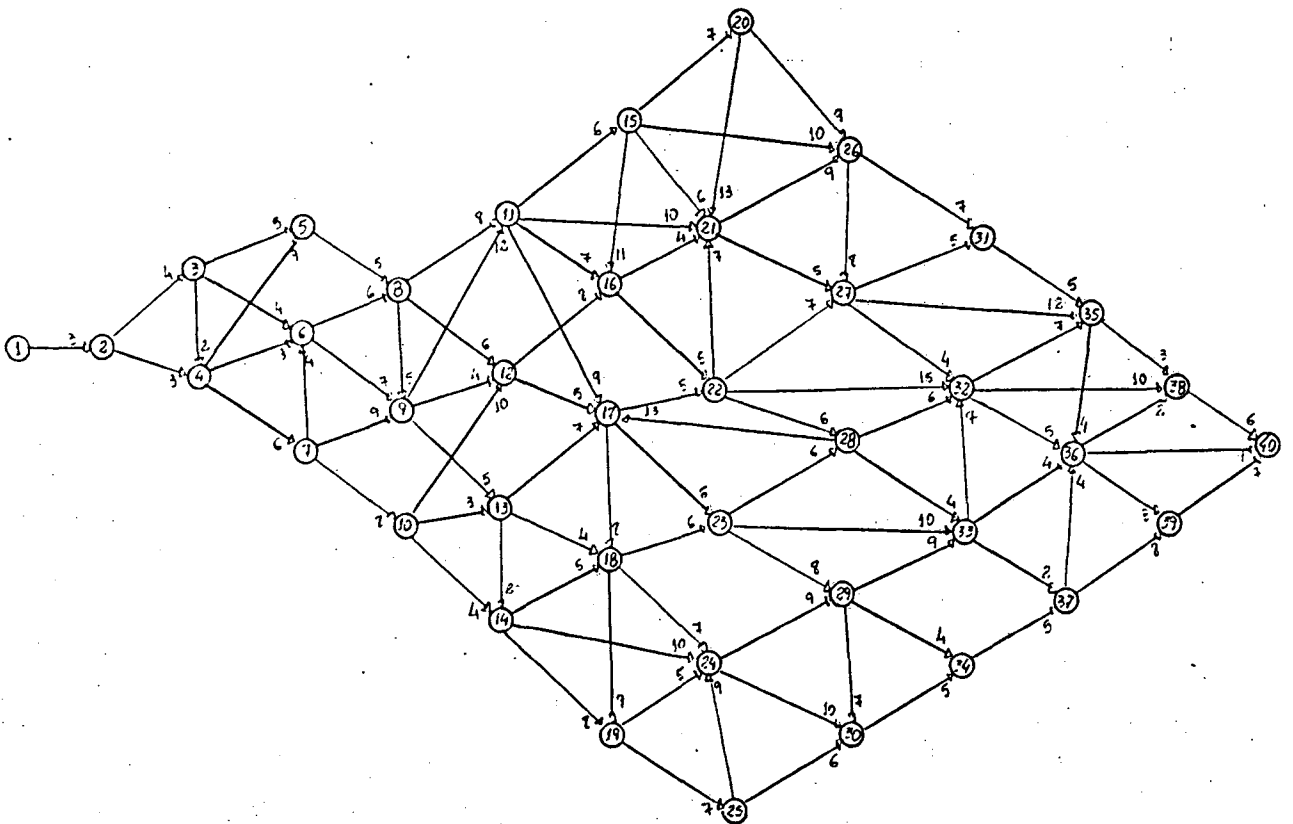
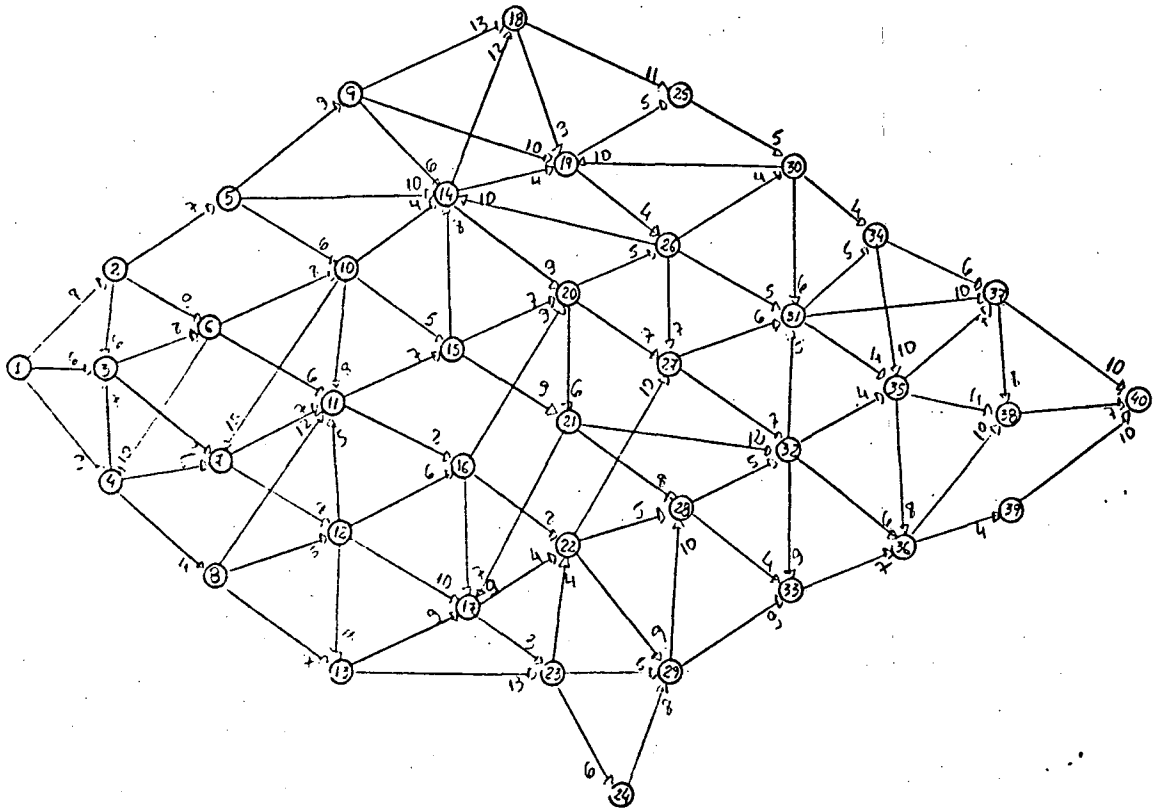






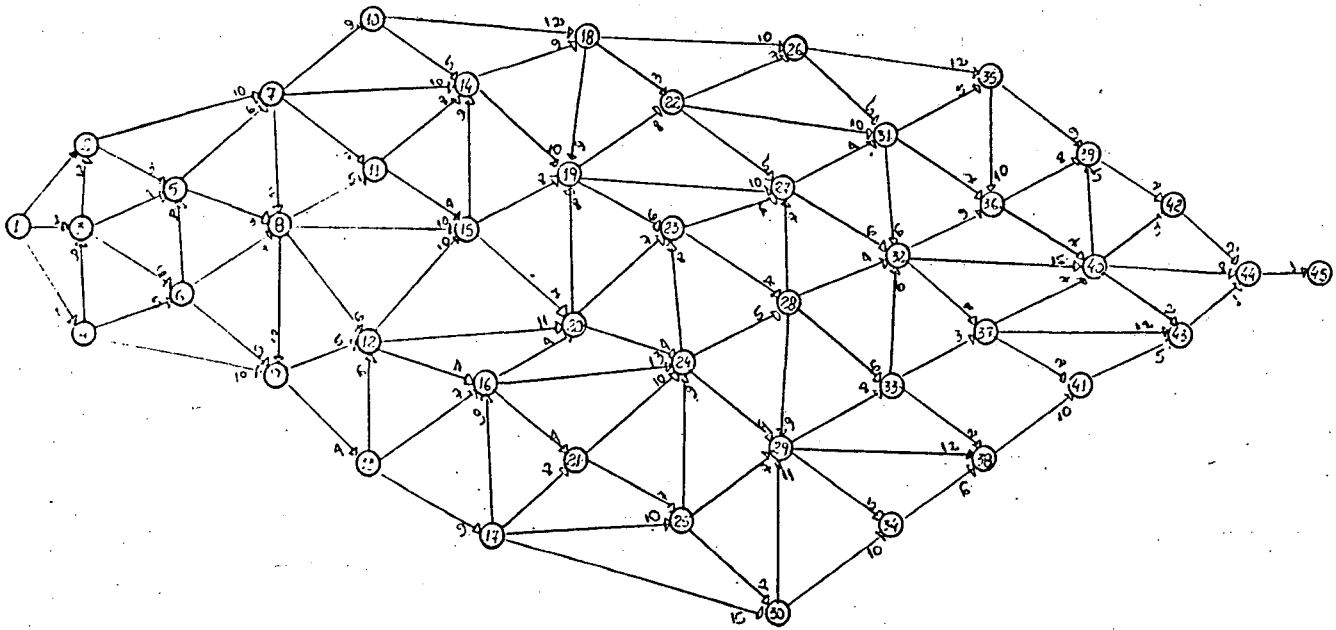
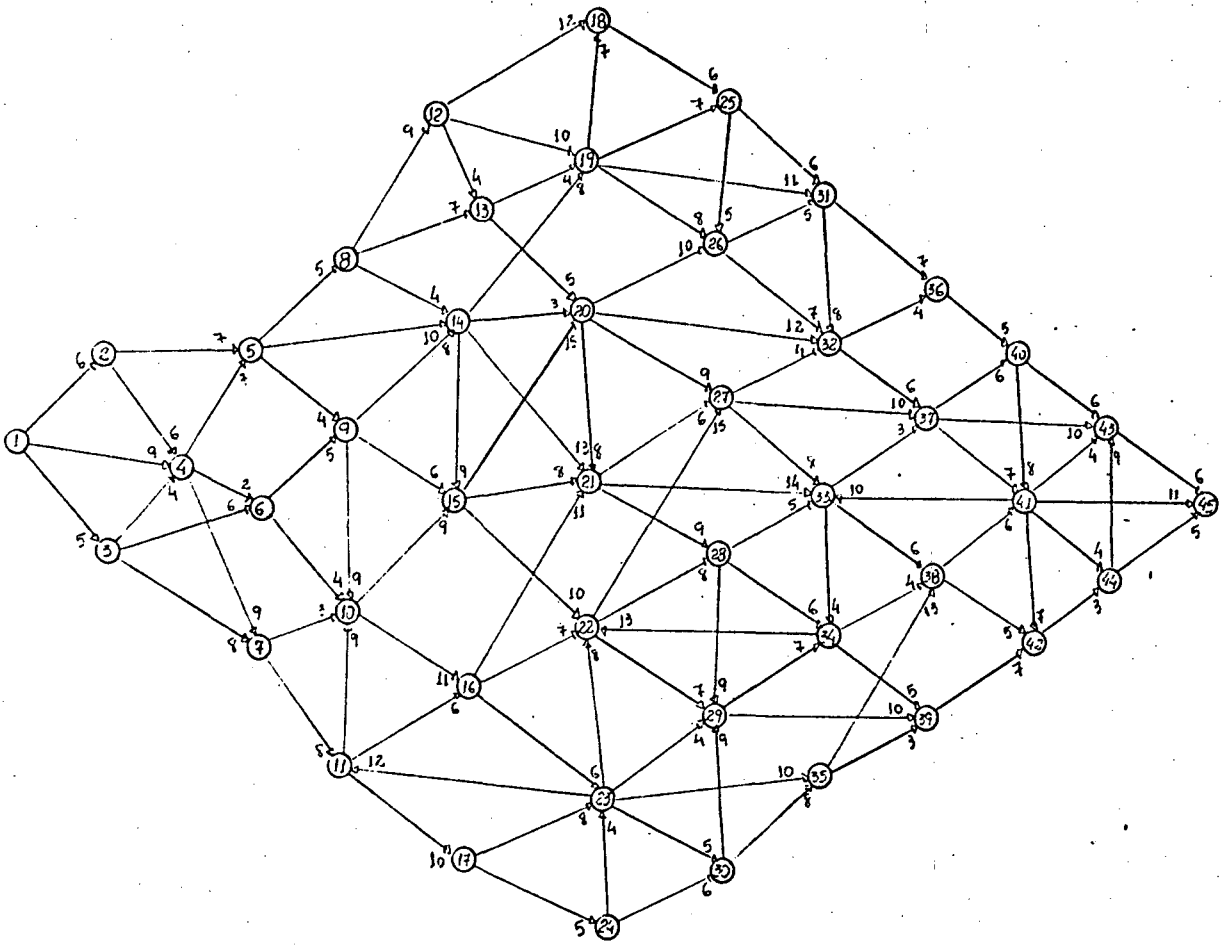


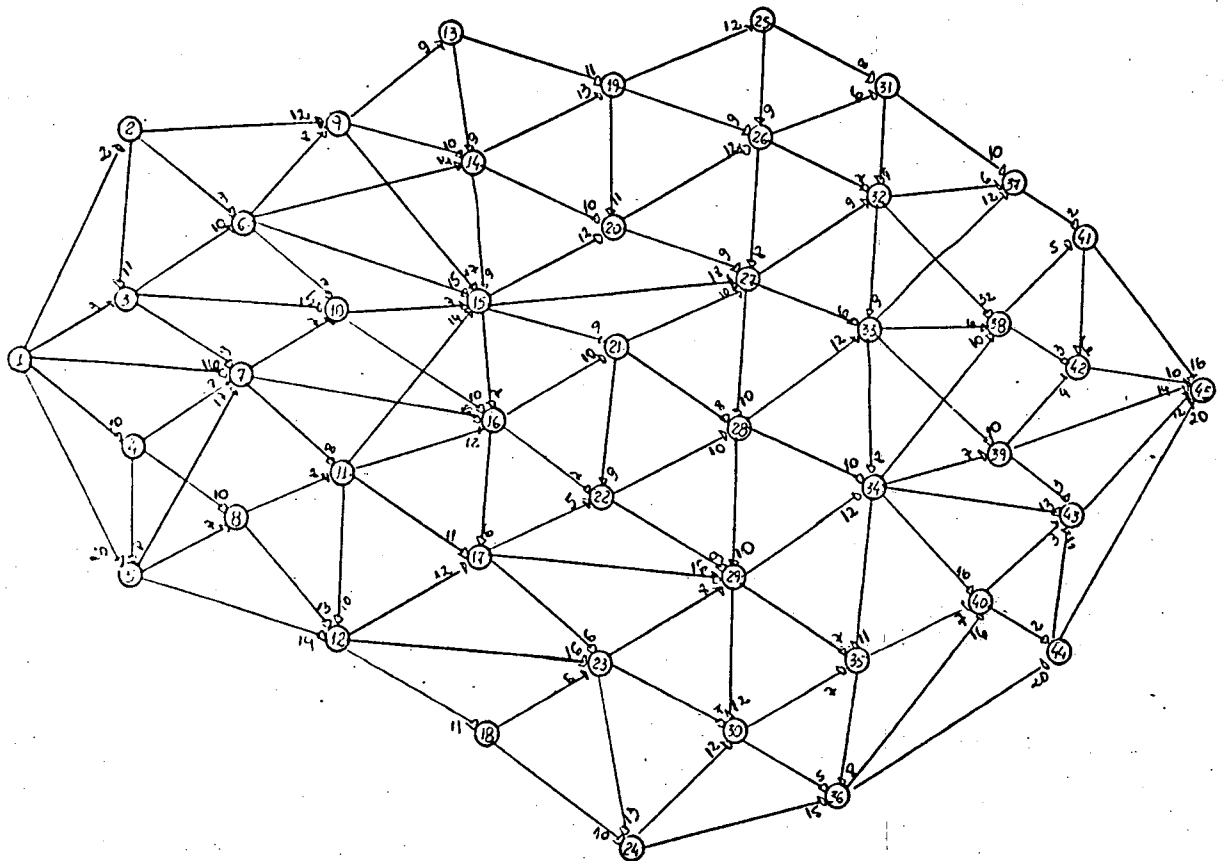
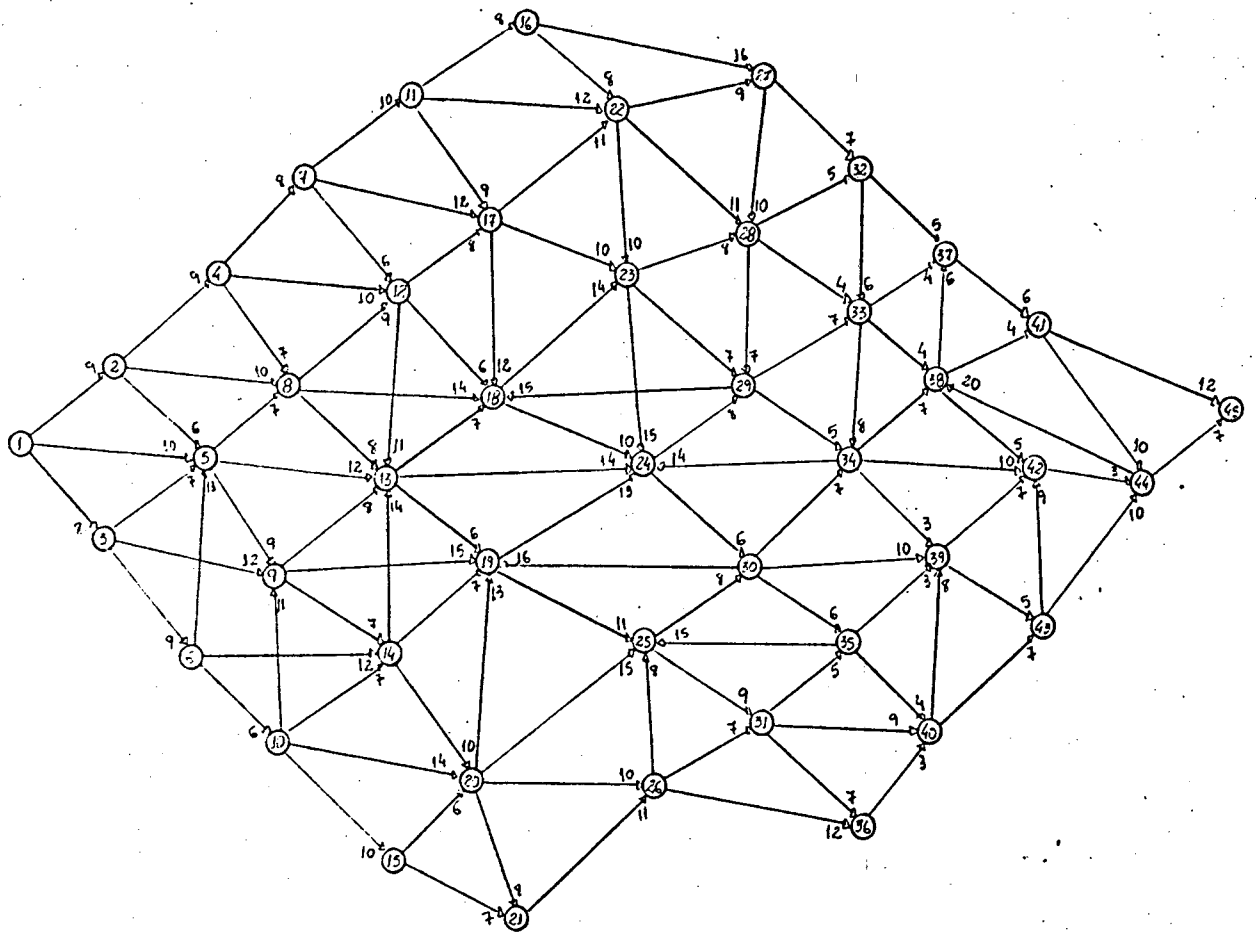


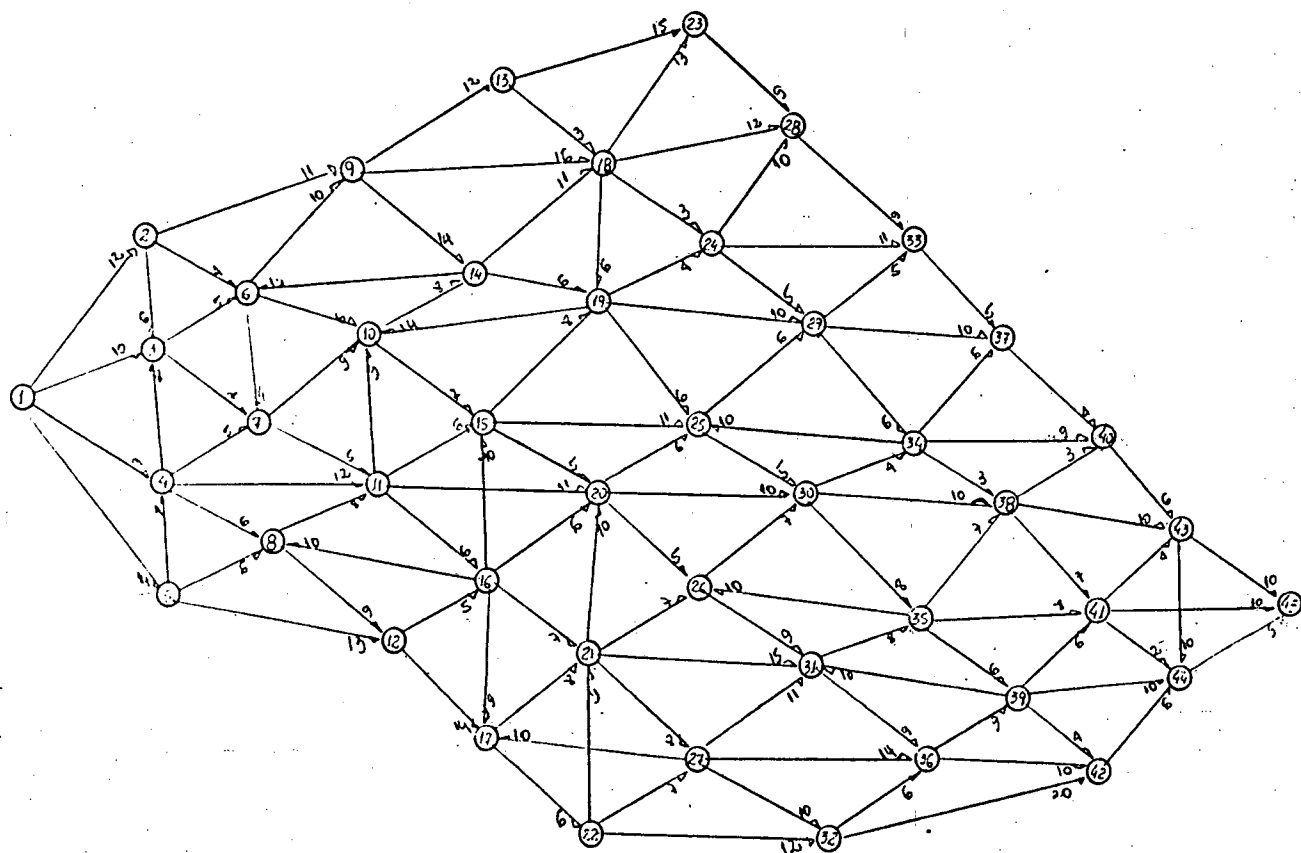
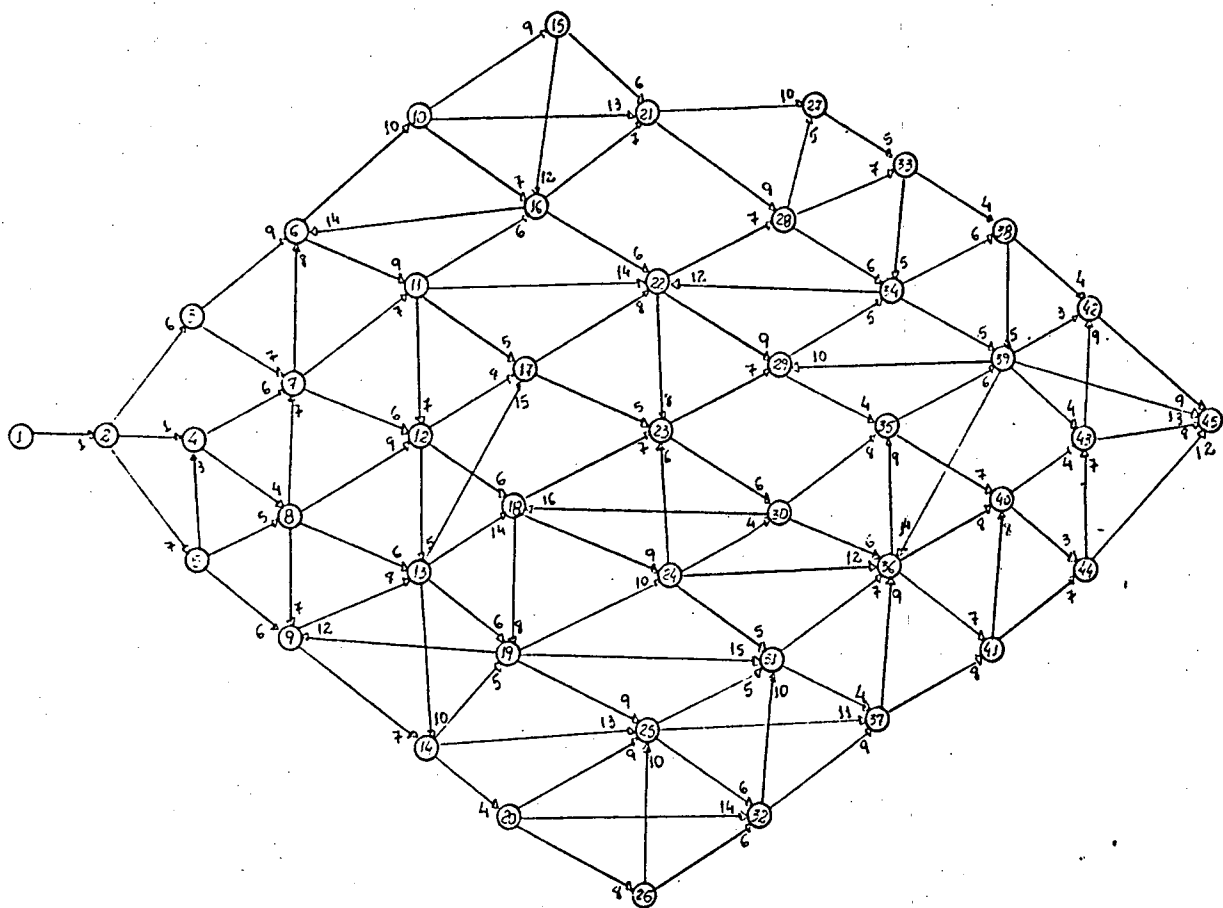


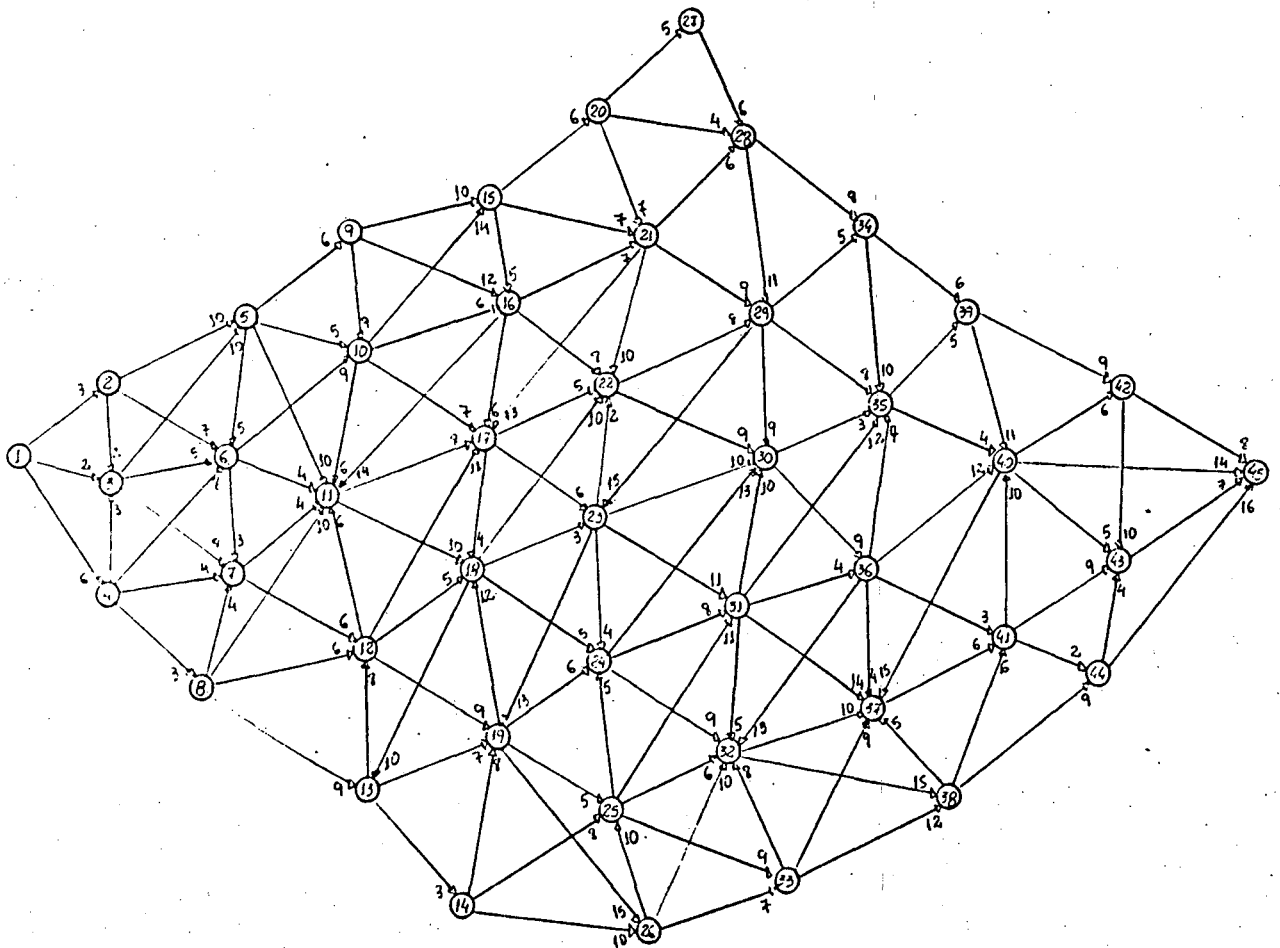
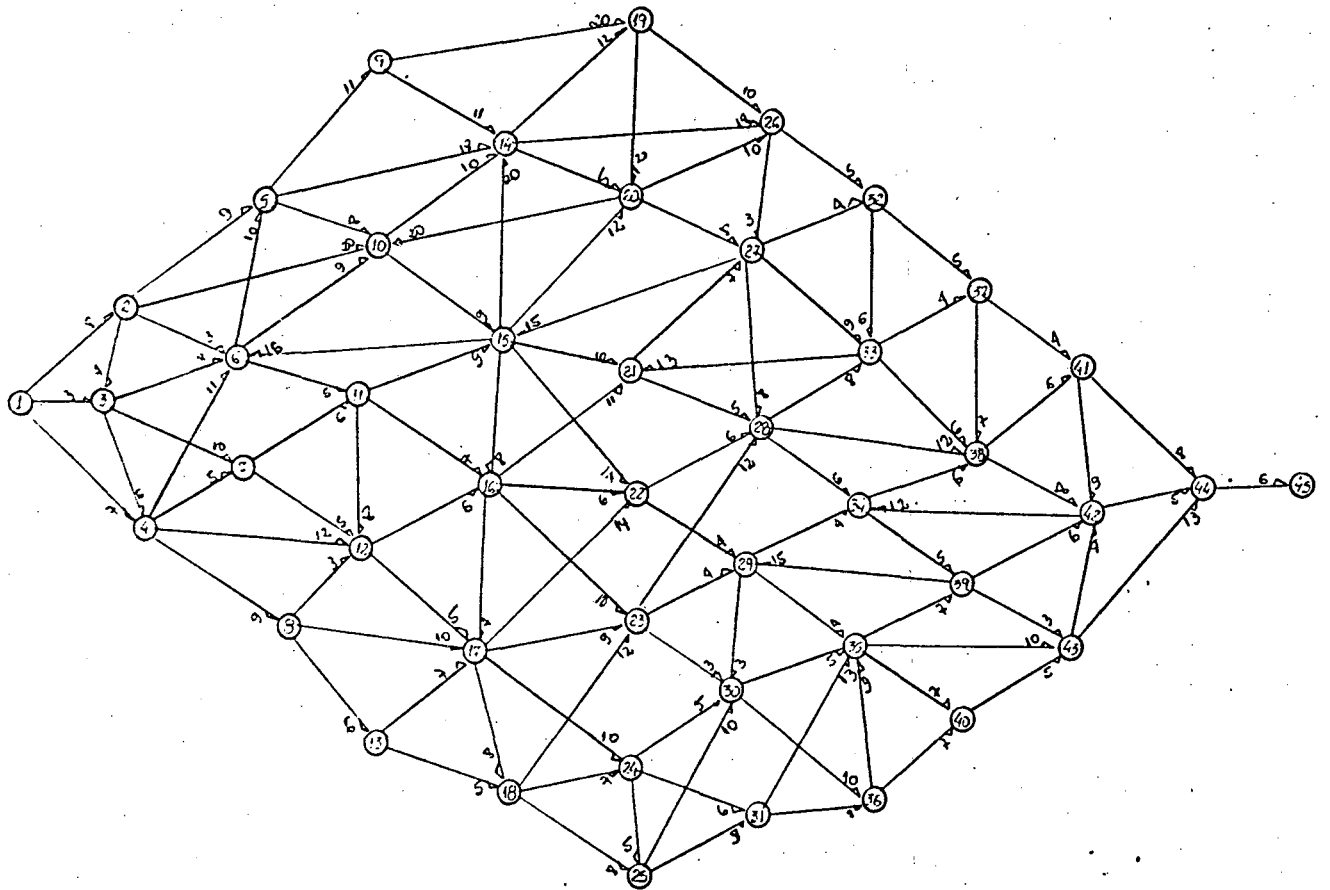
F I G U R A 13

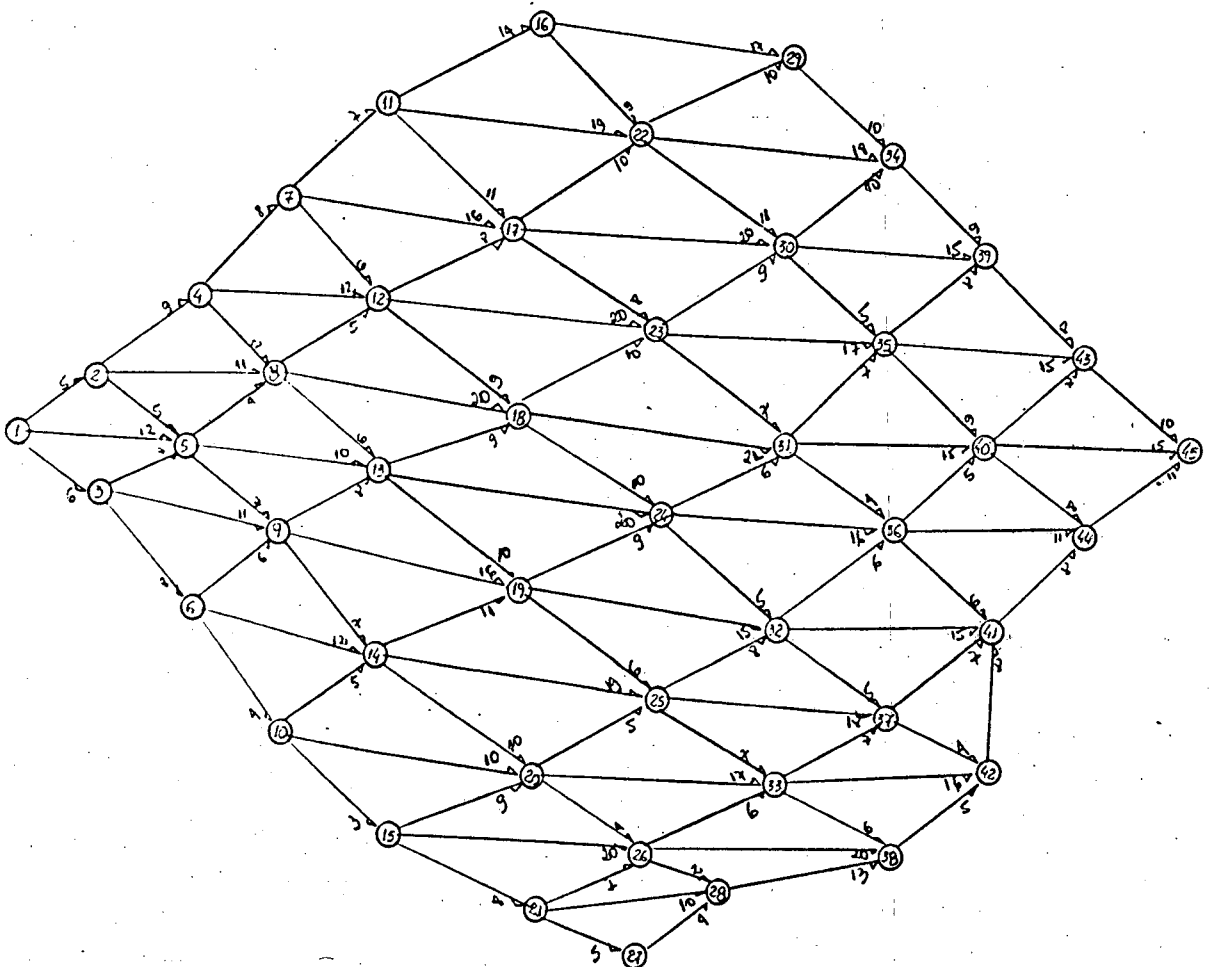
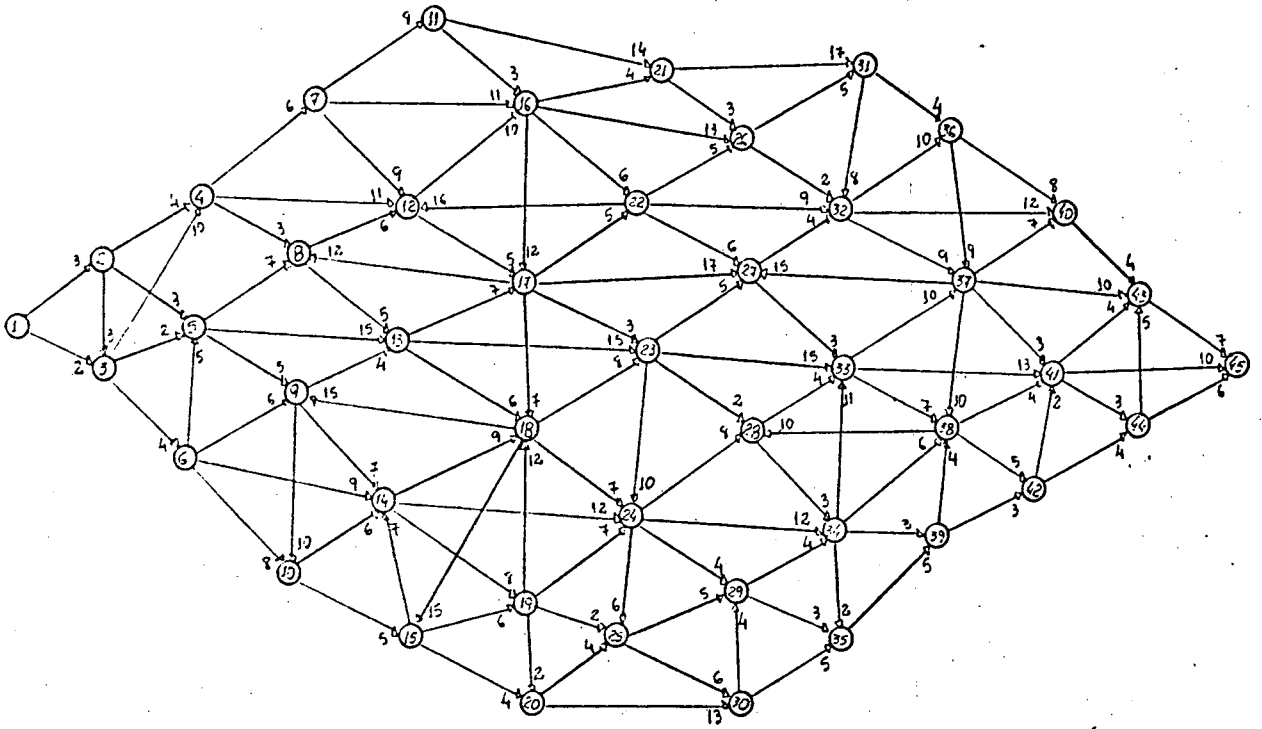
GRAFOS COM 45 VÉRTICES

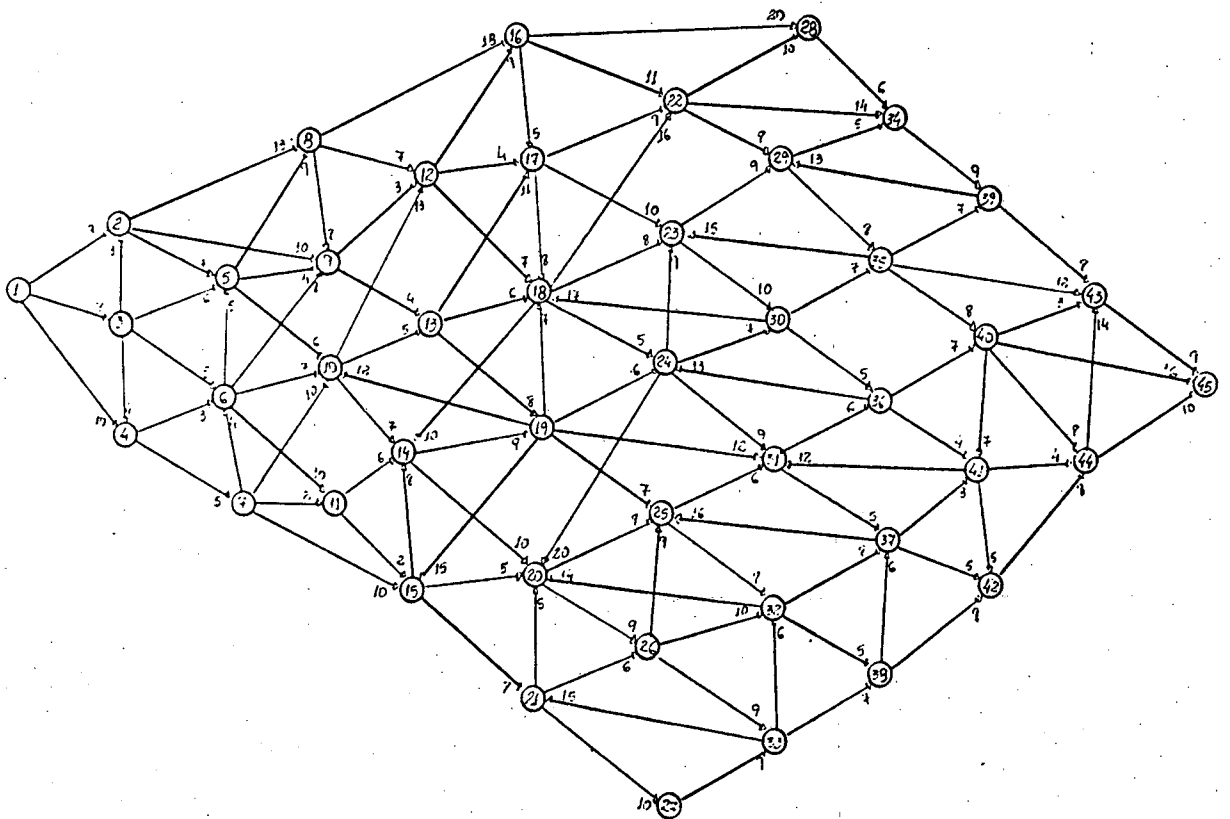
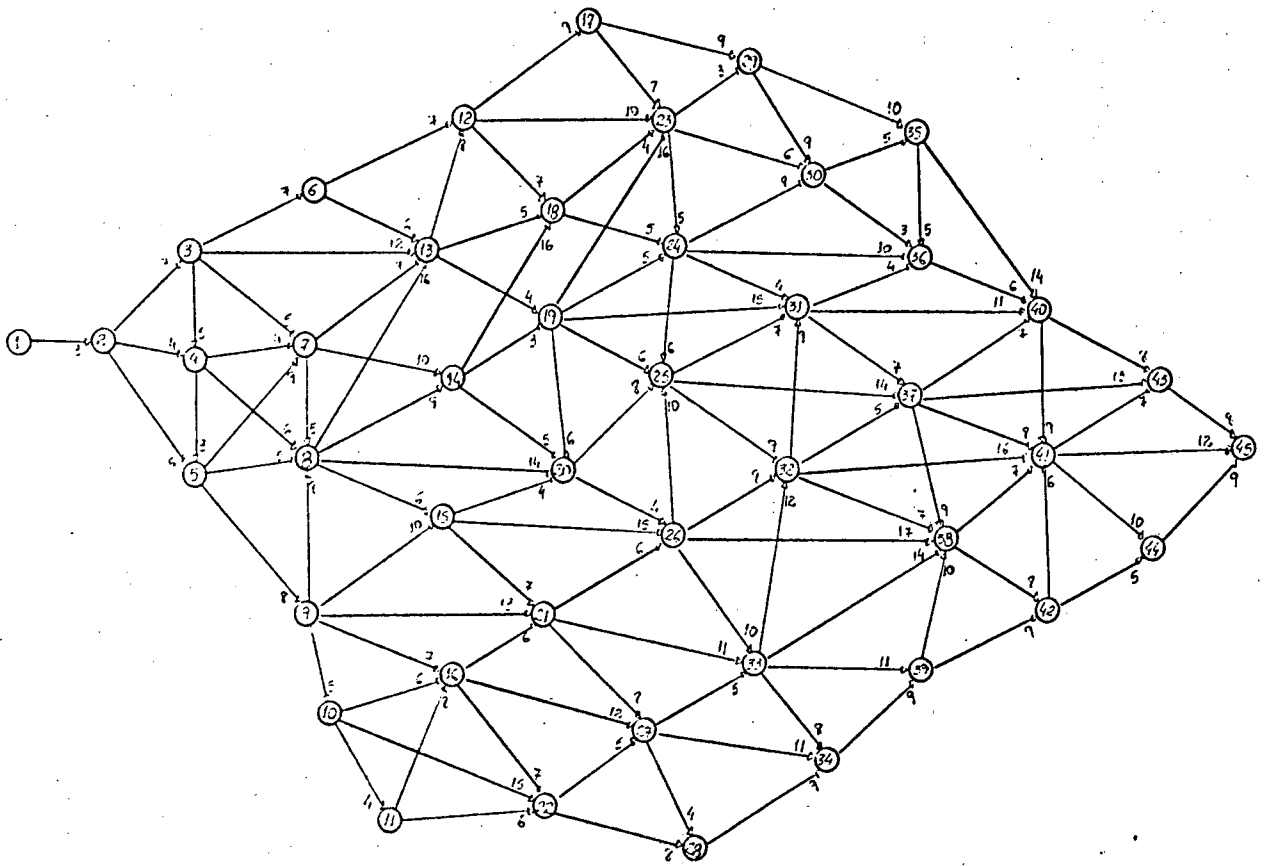


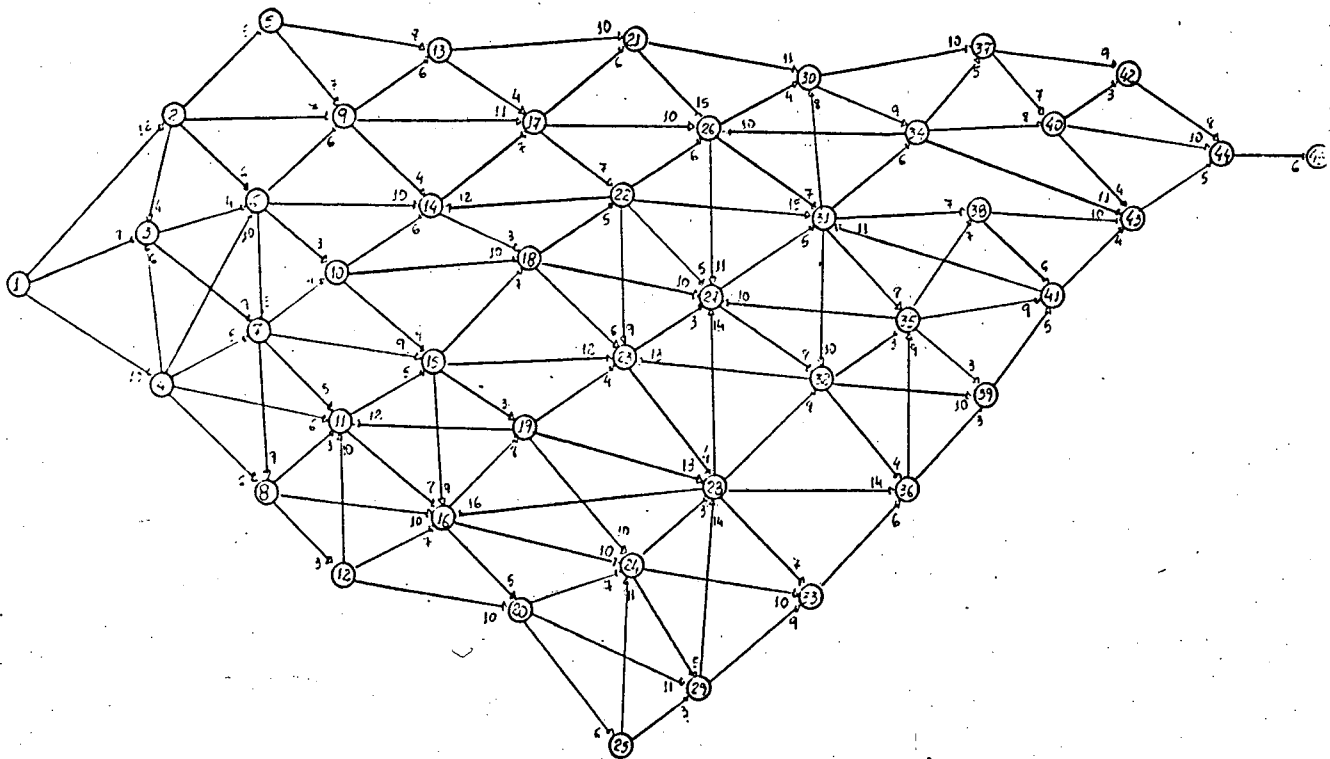
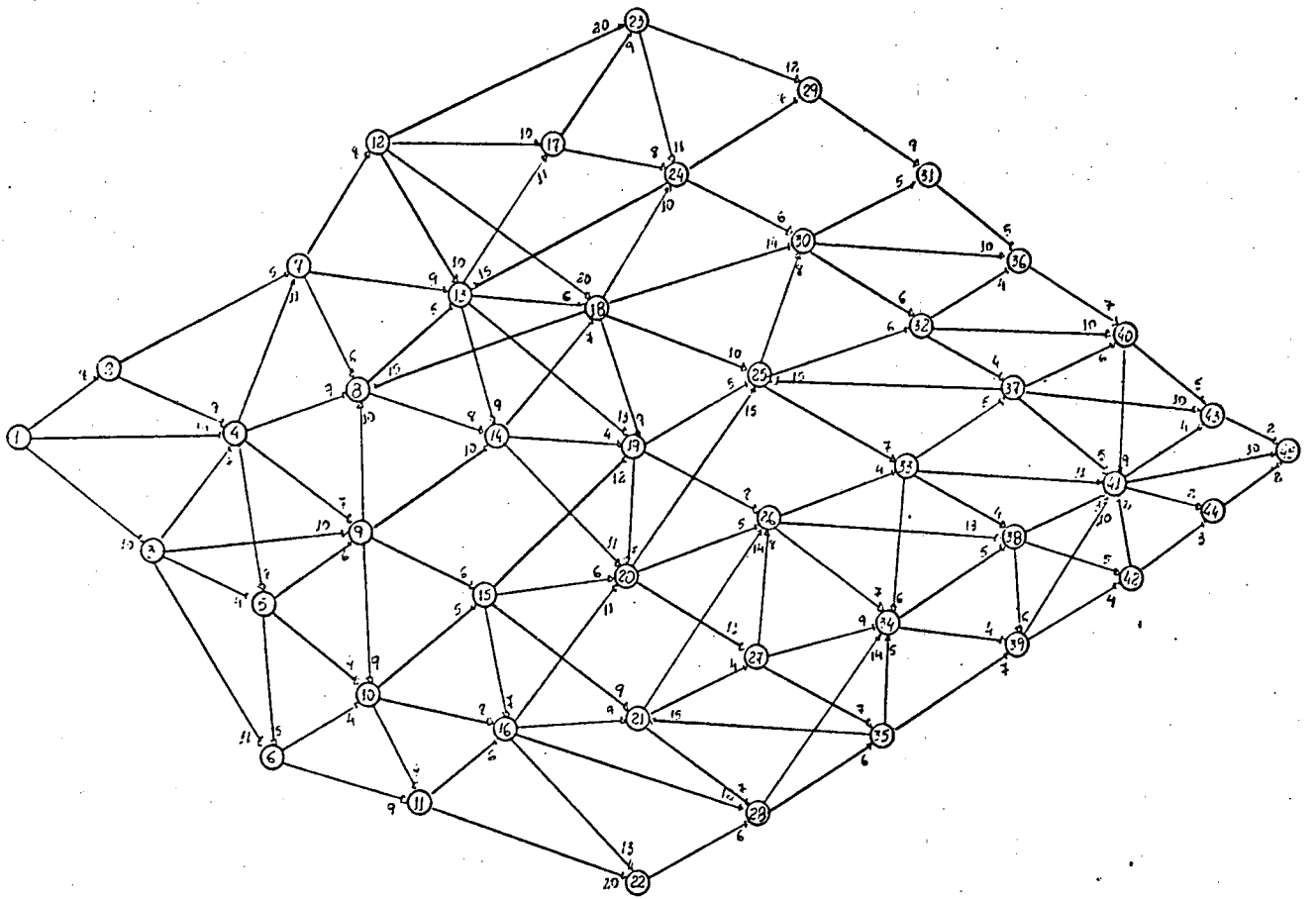


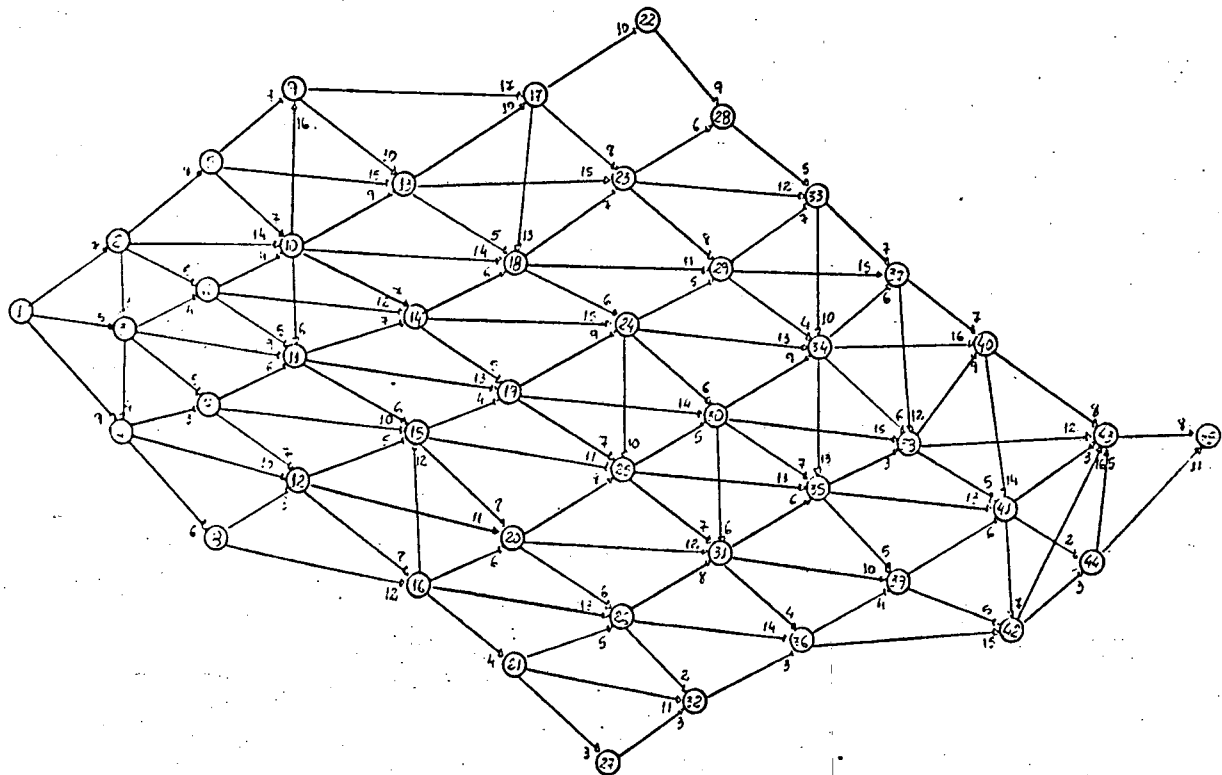
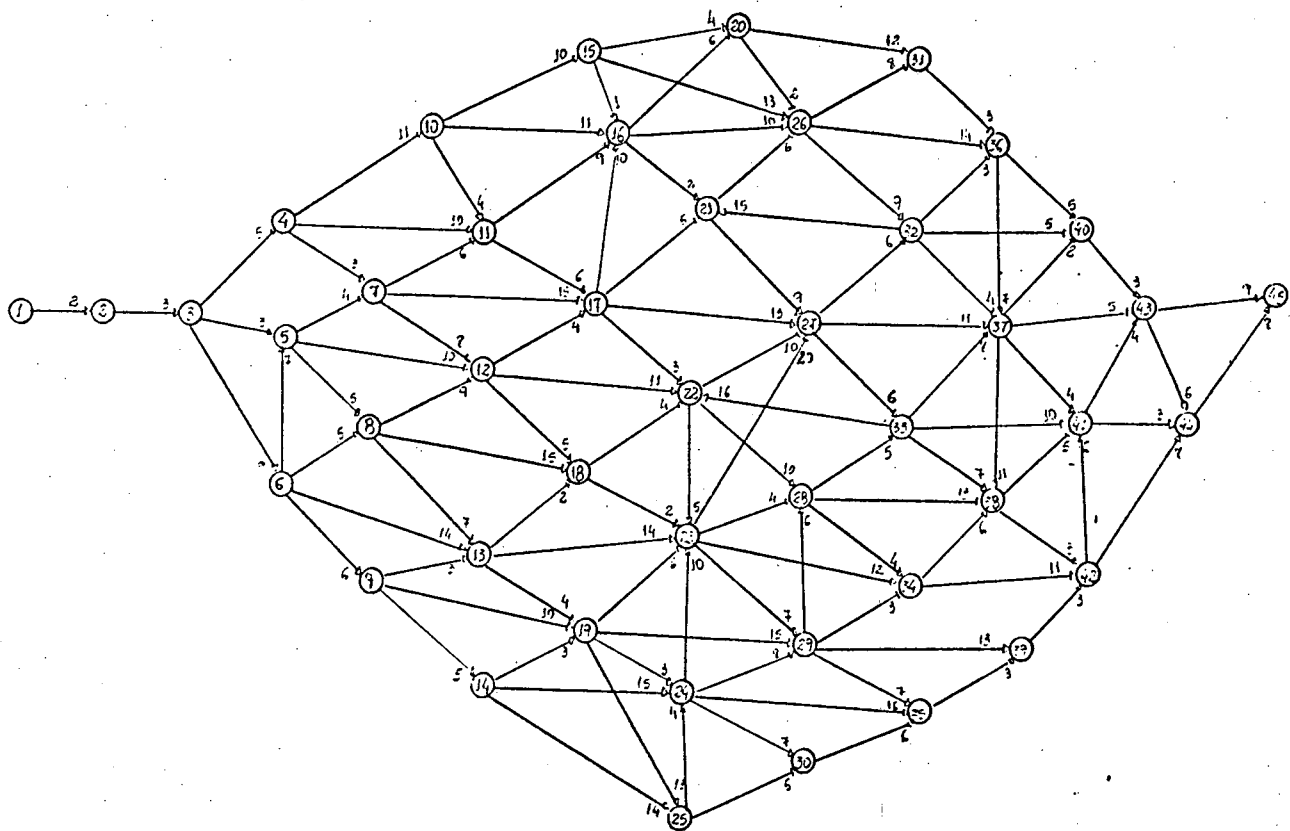


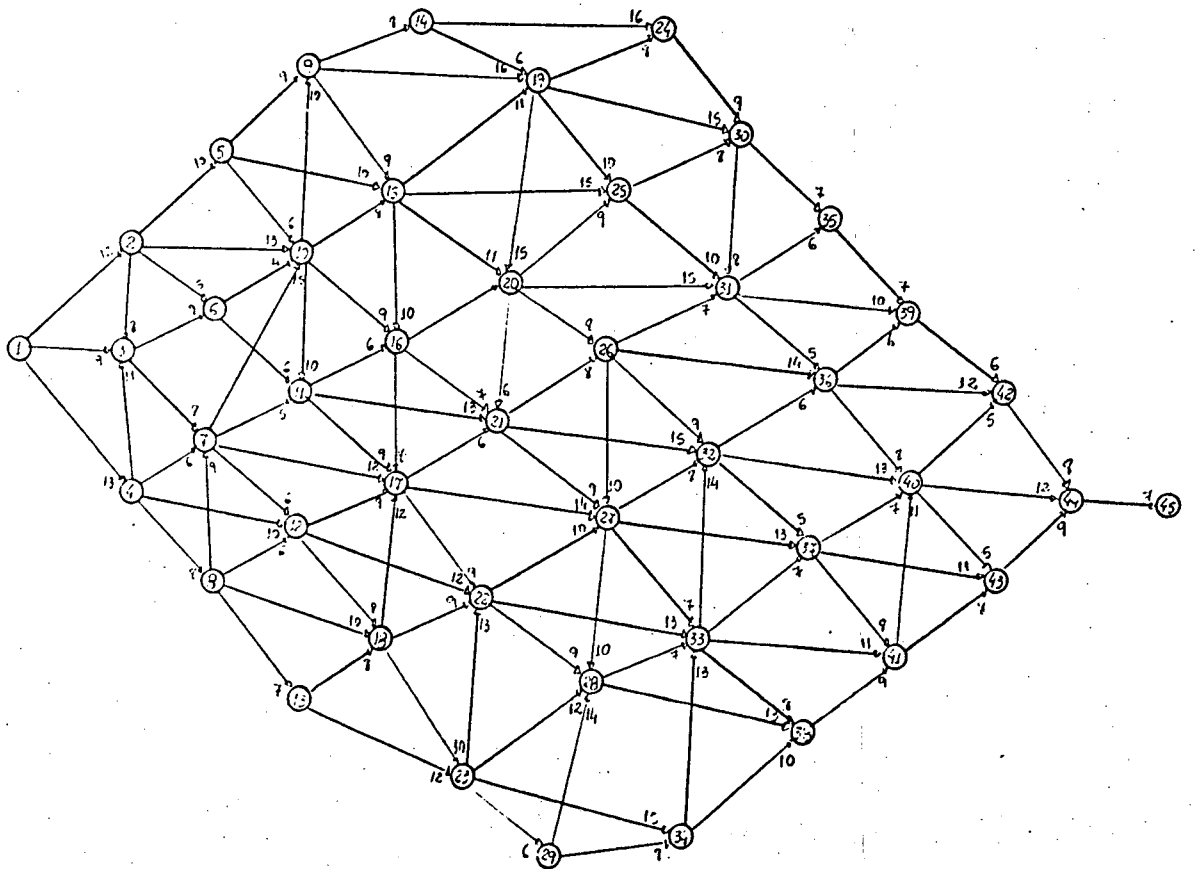
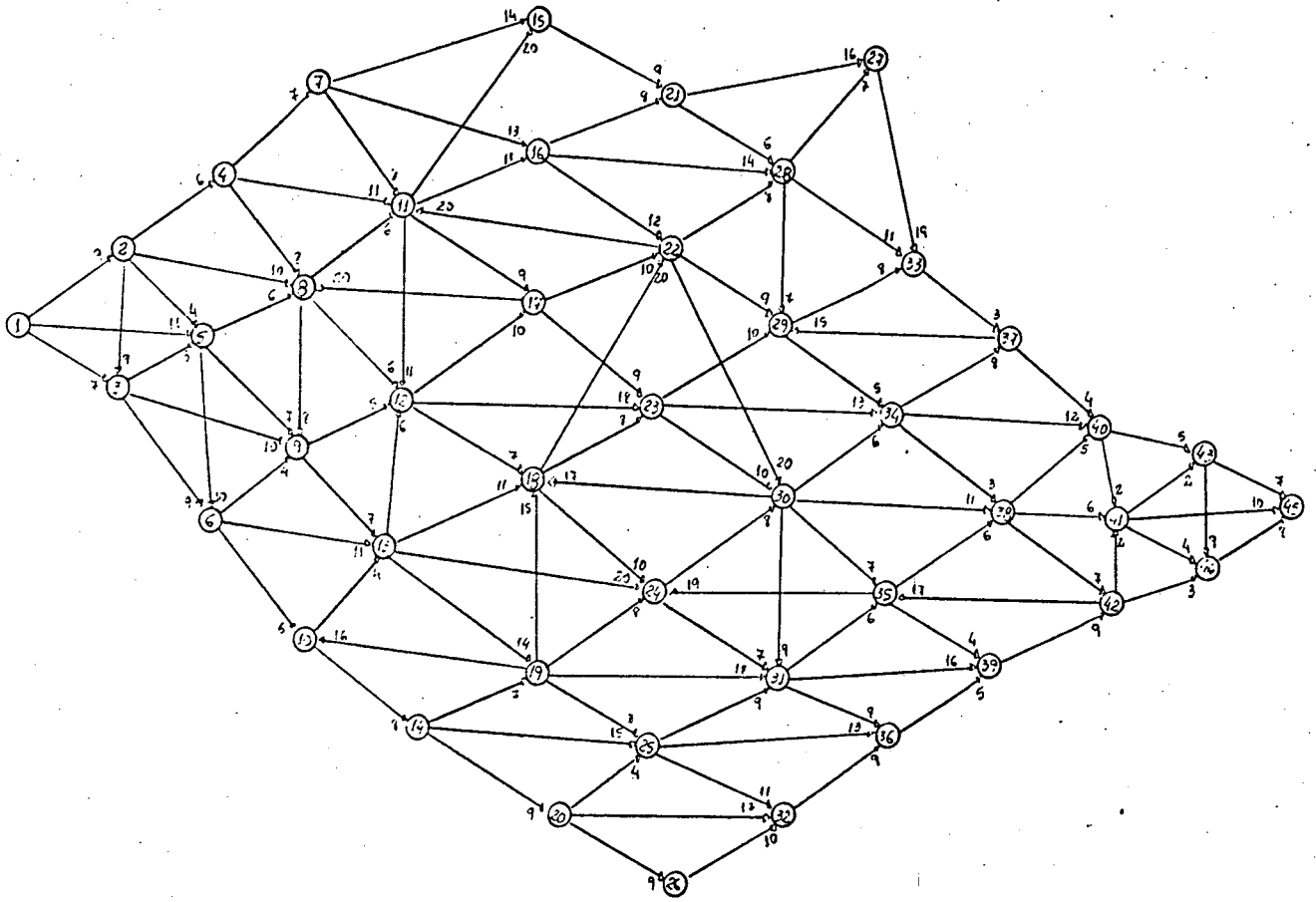


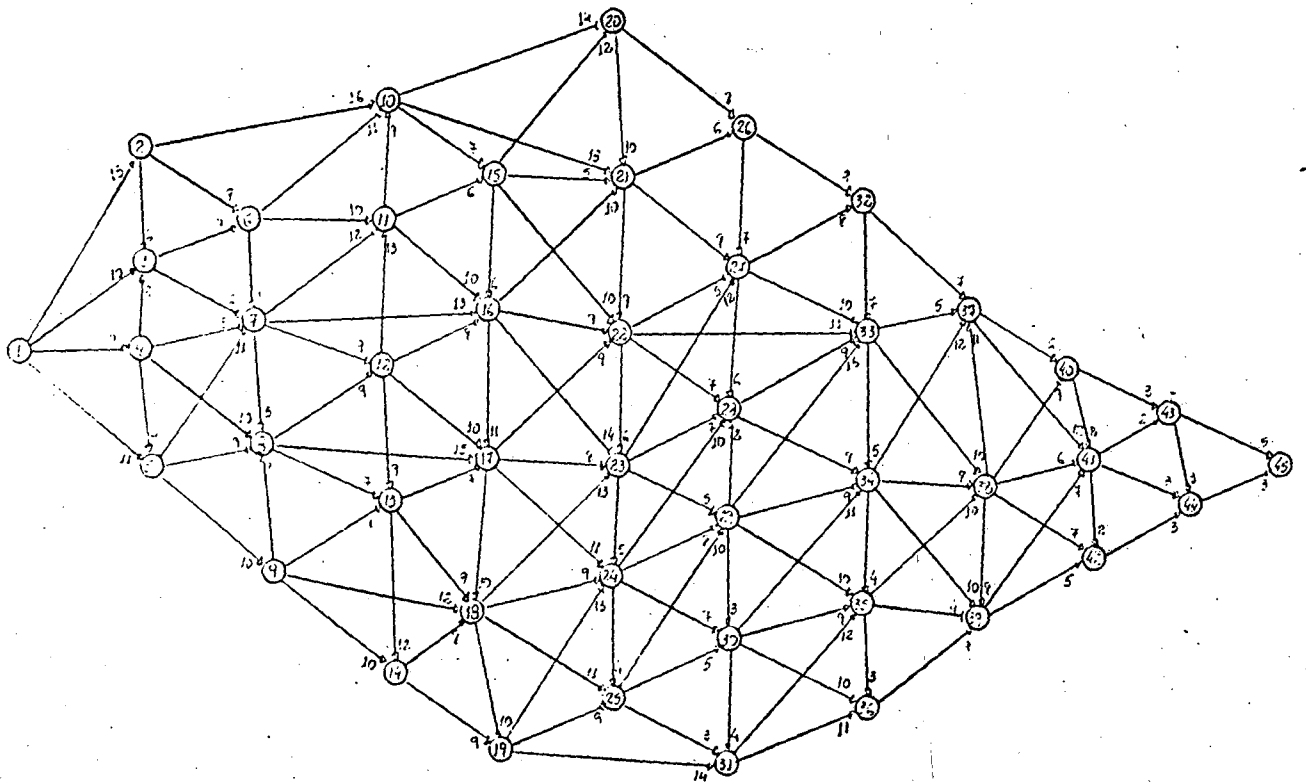
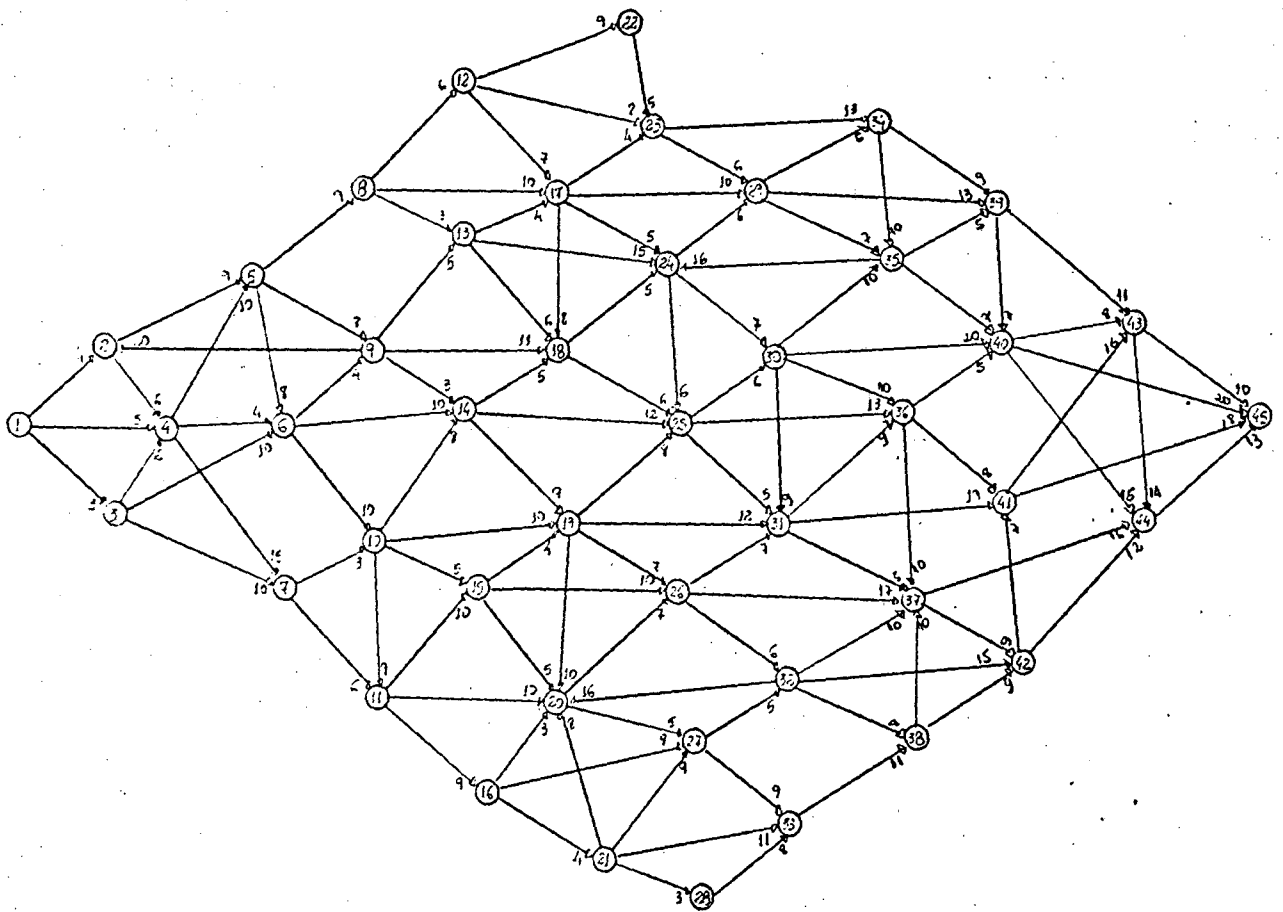






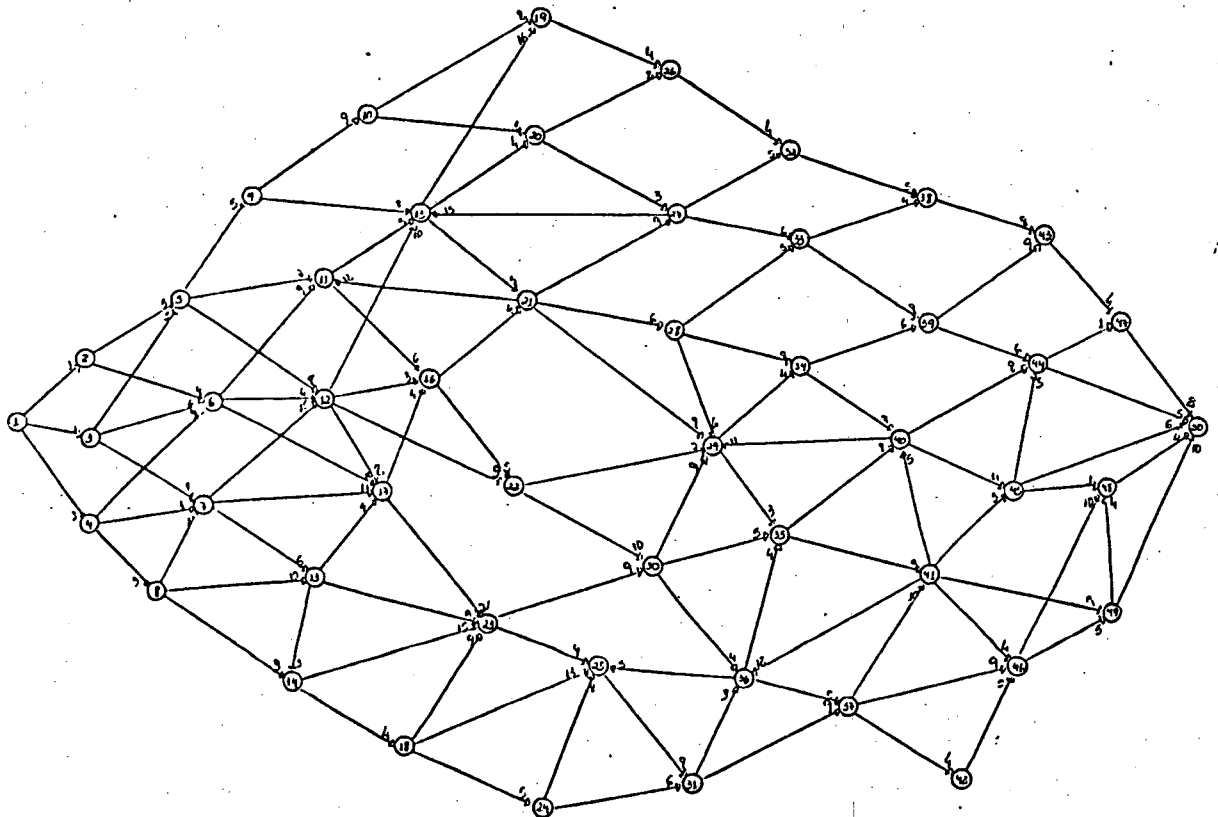
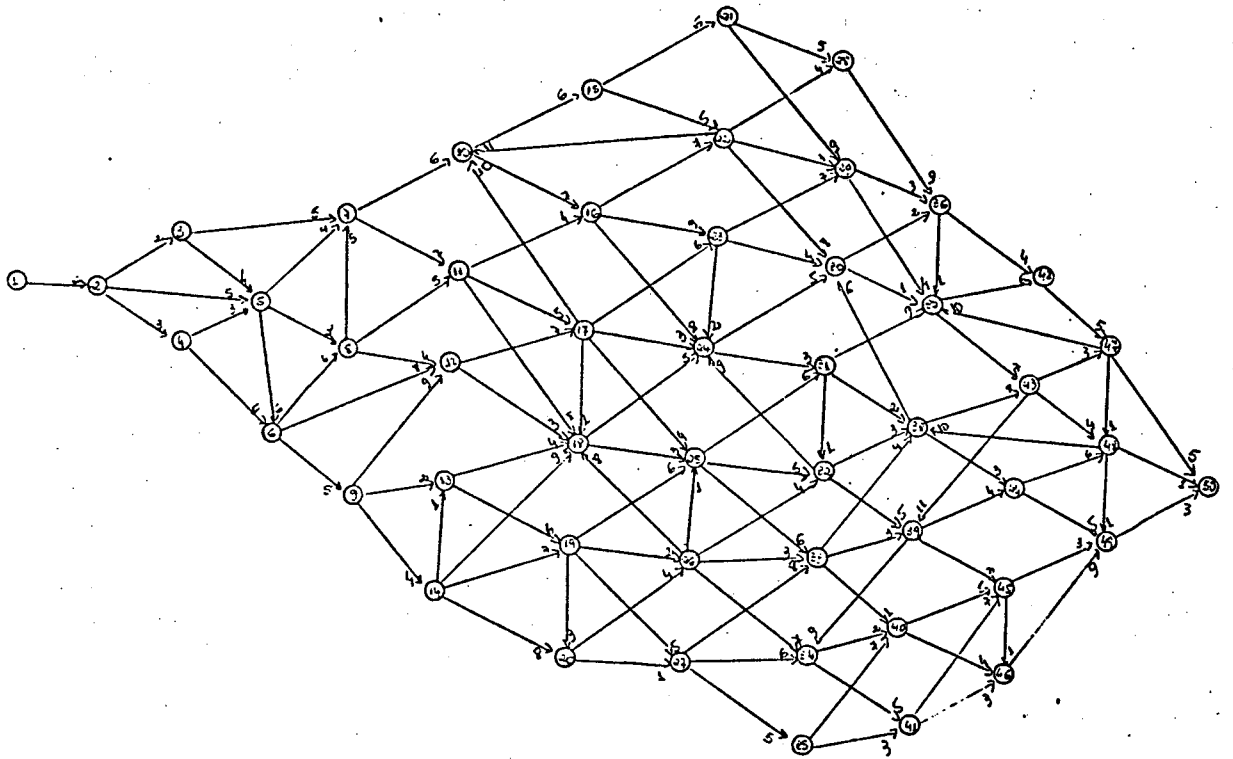


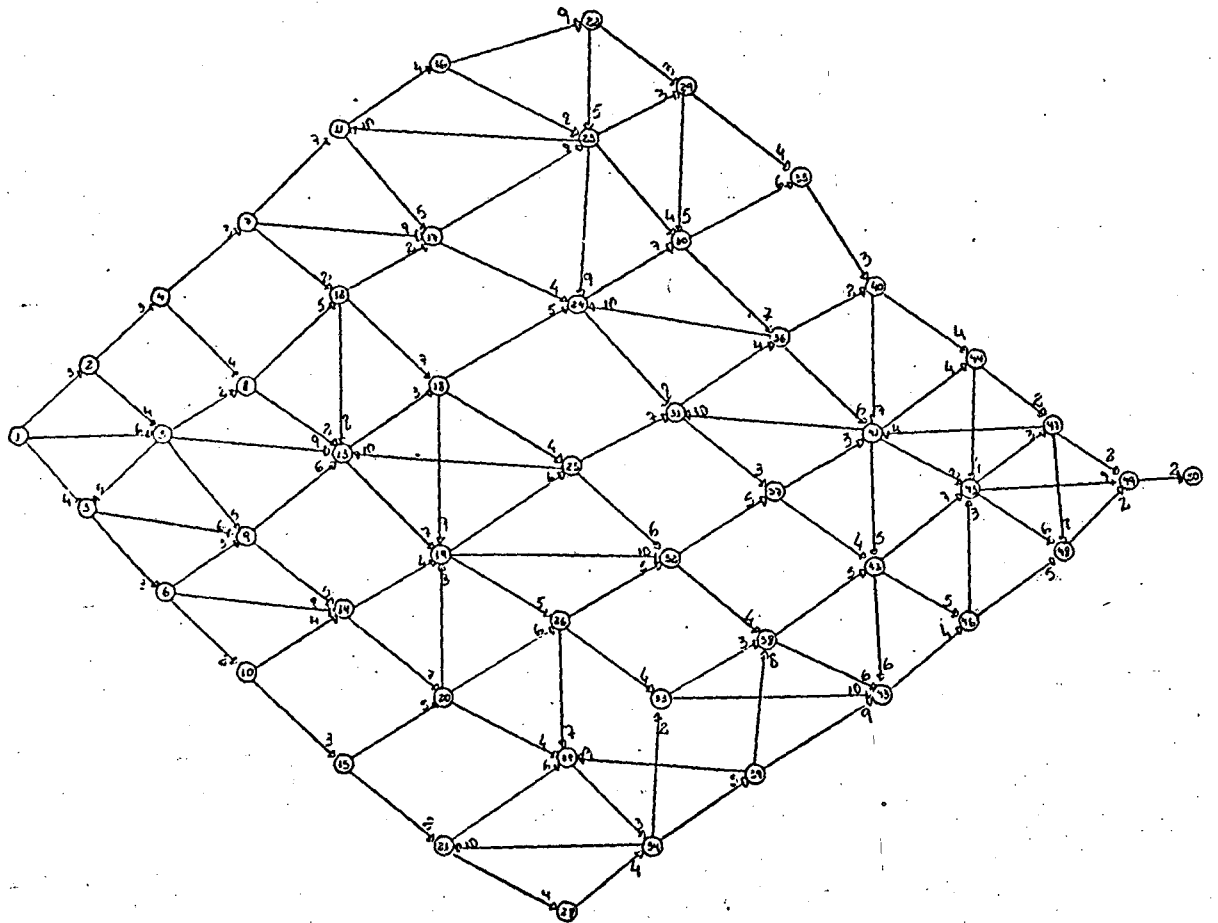
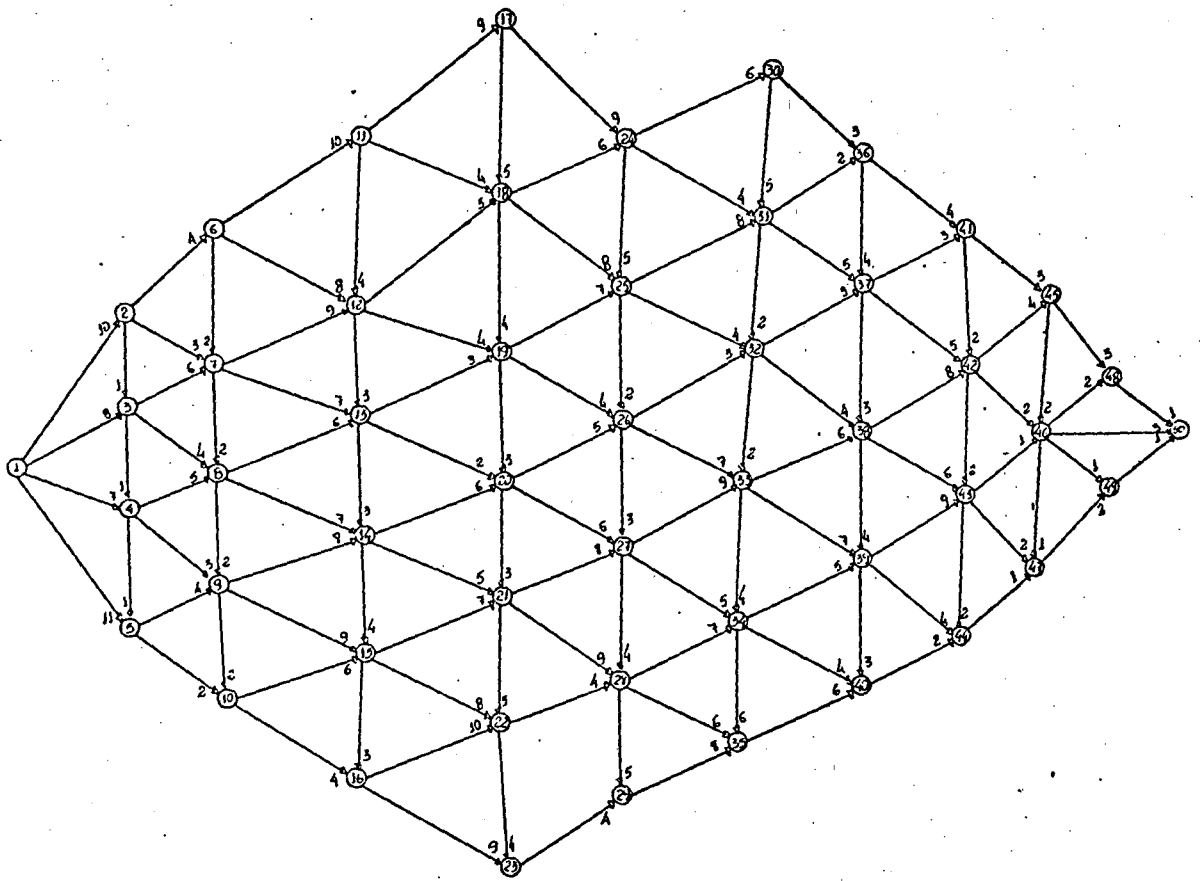


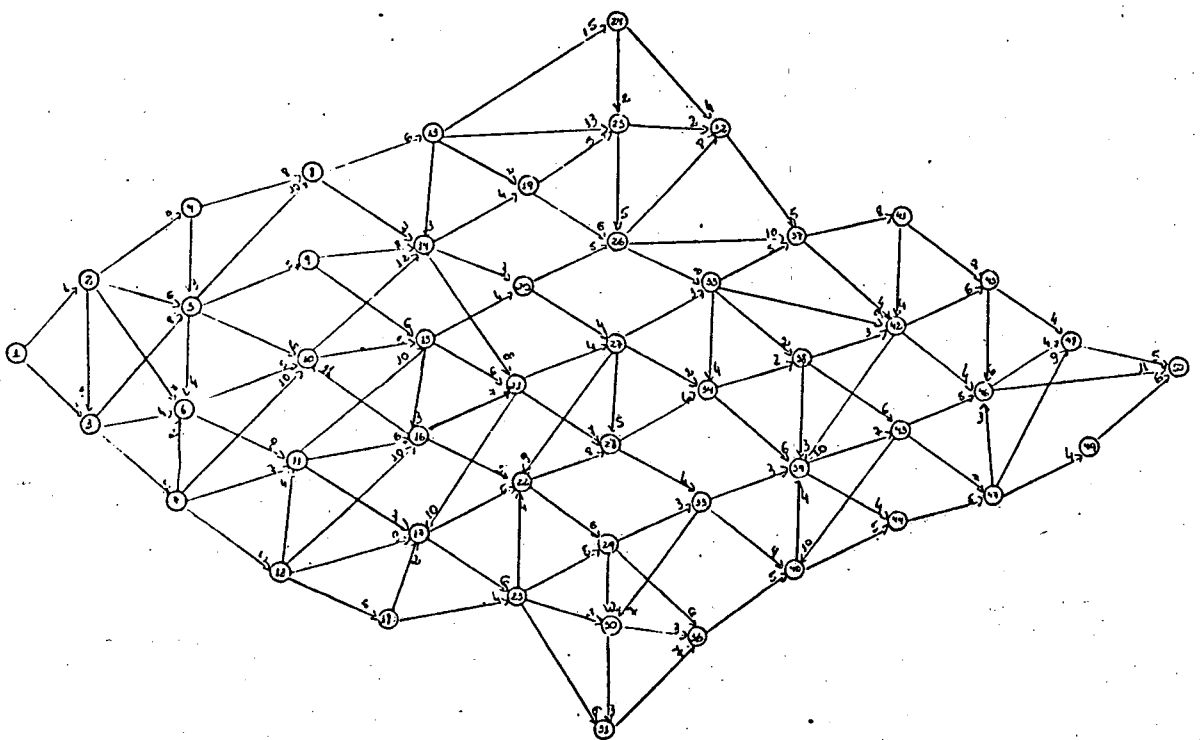
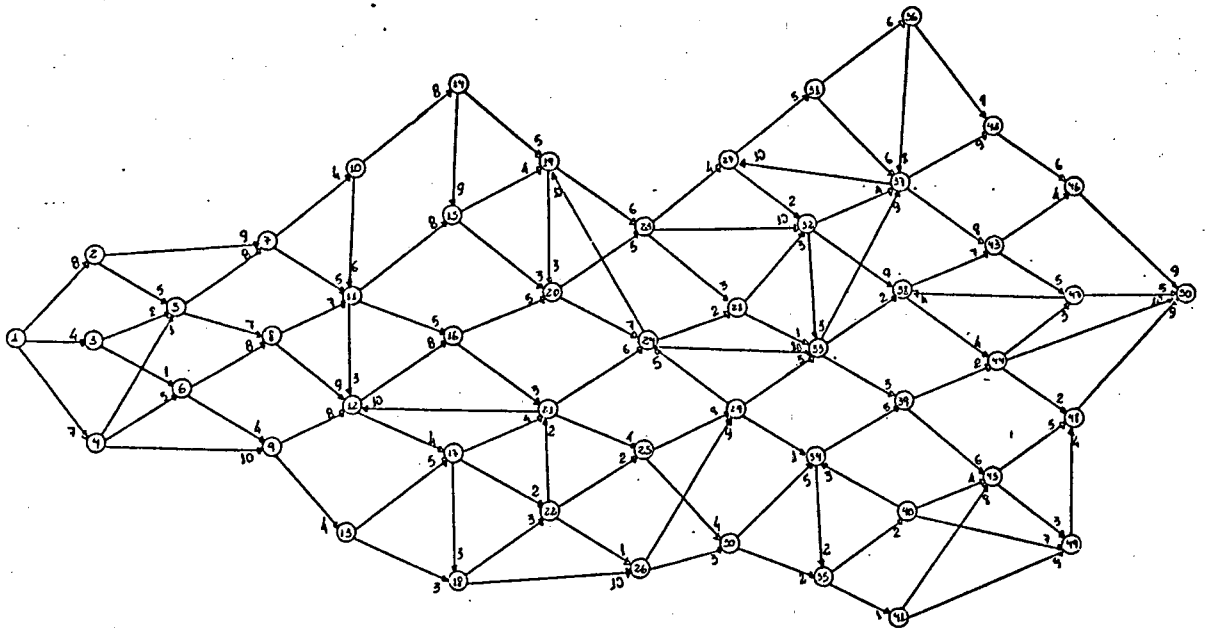


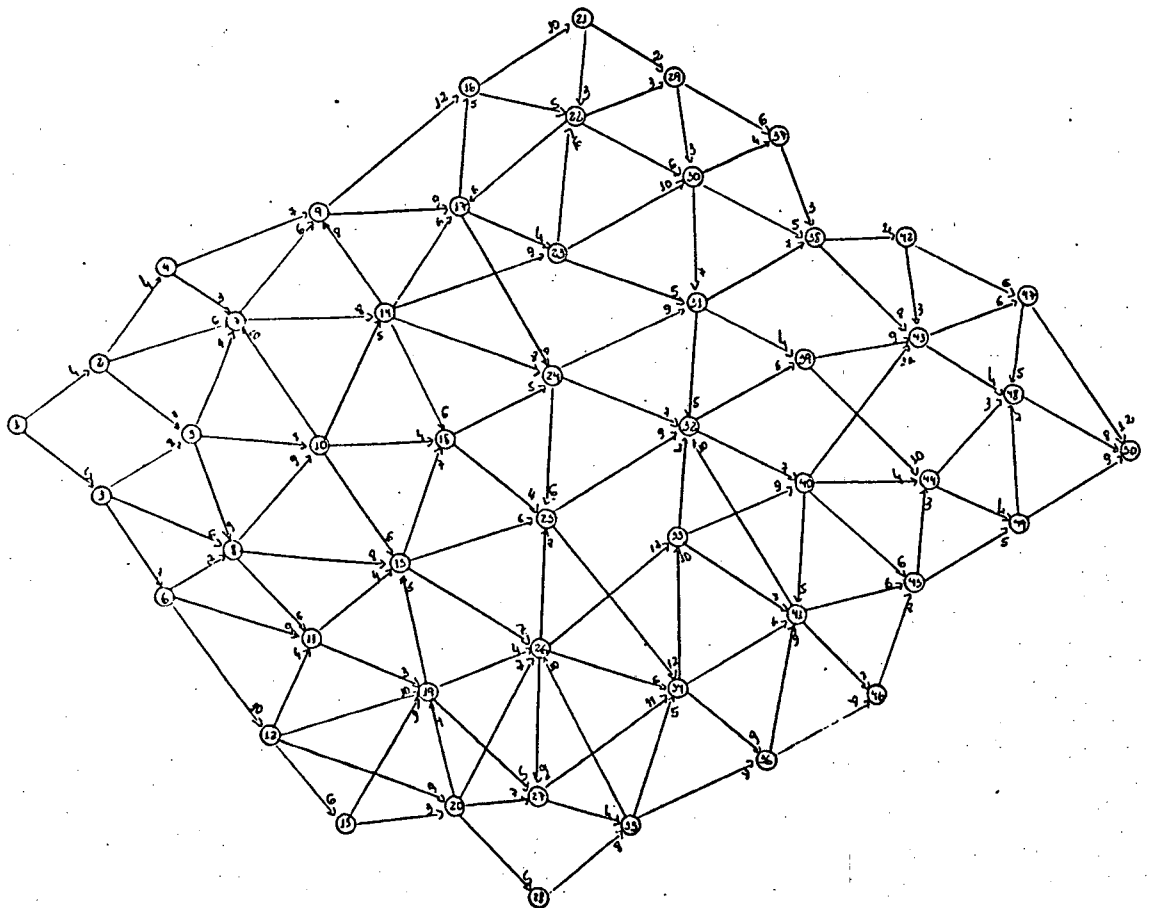
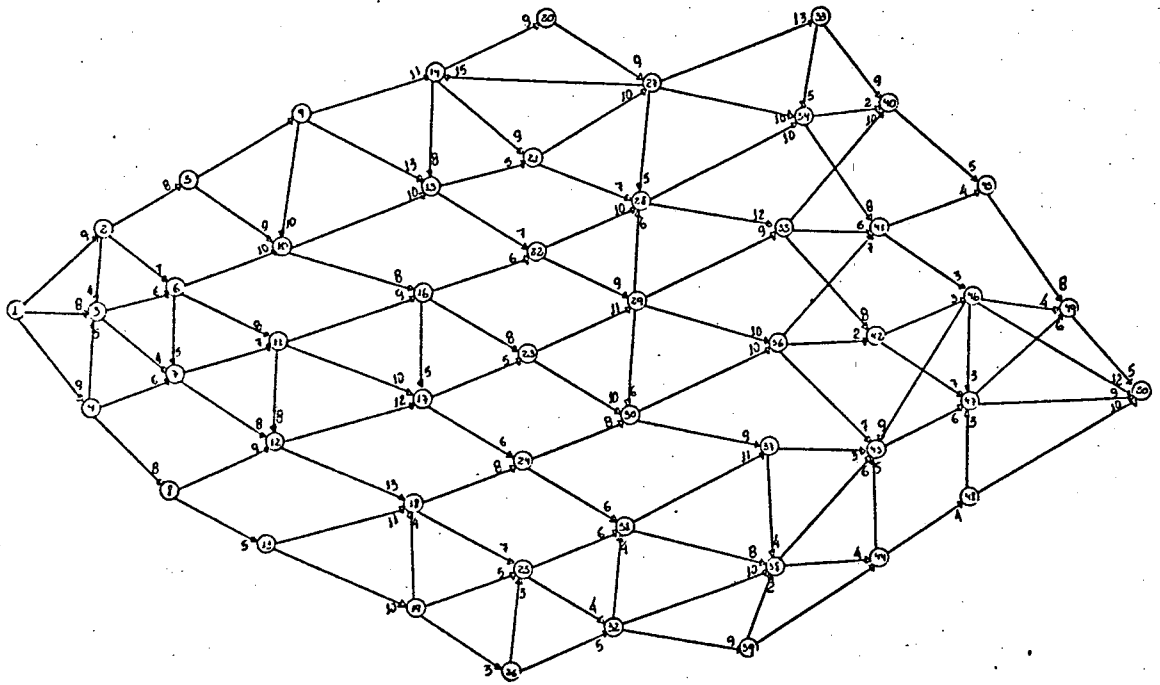
F I G U R A 14

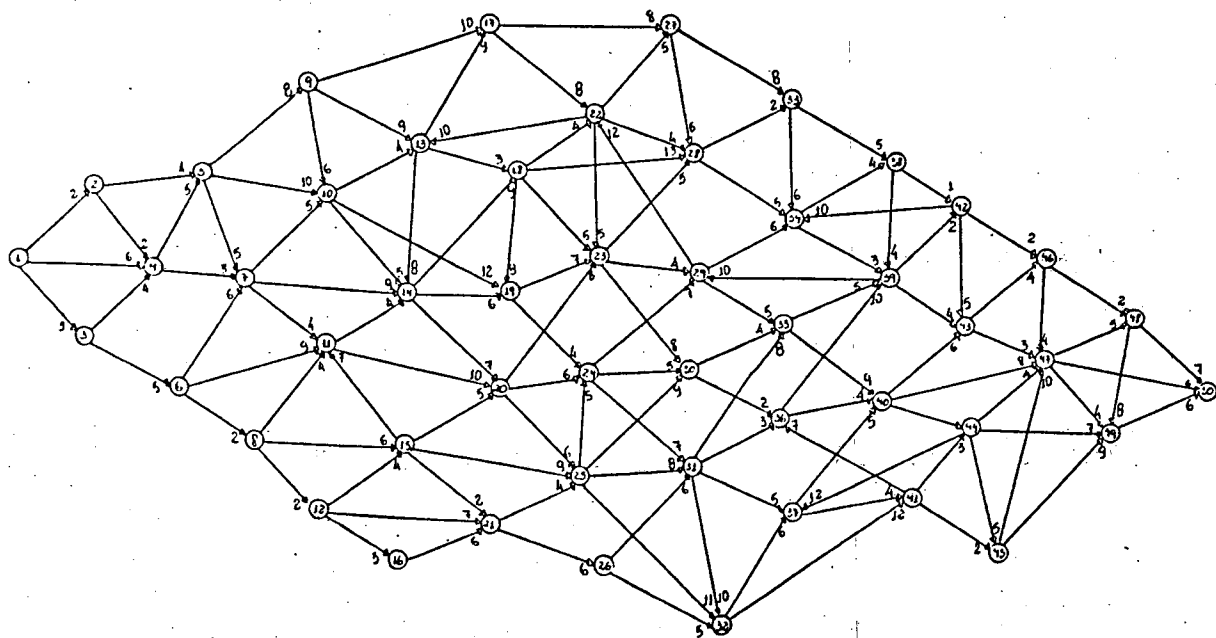
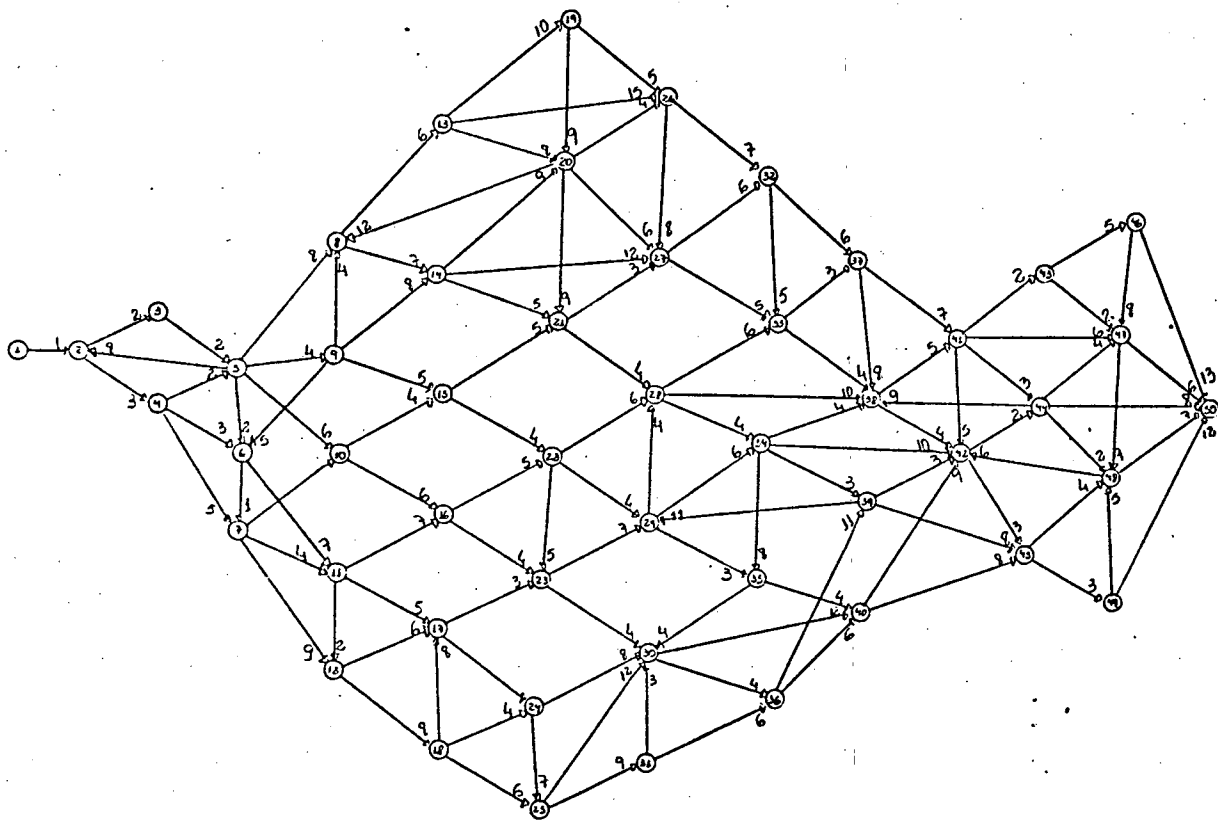
GRAFOS COM 50 VÉRTICES

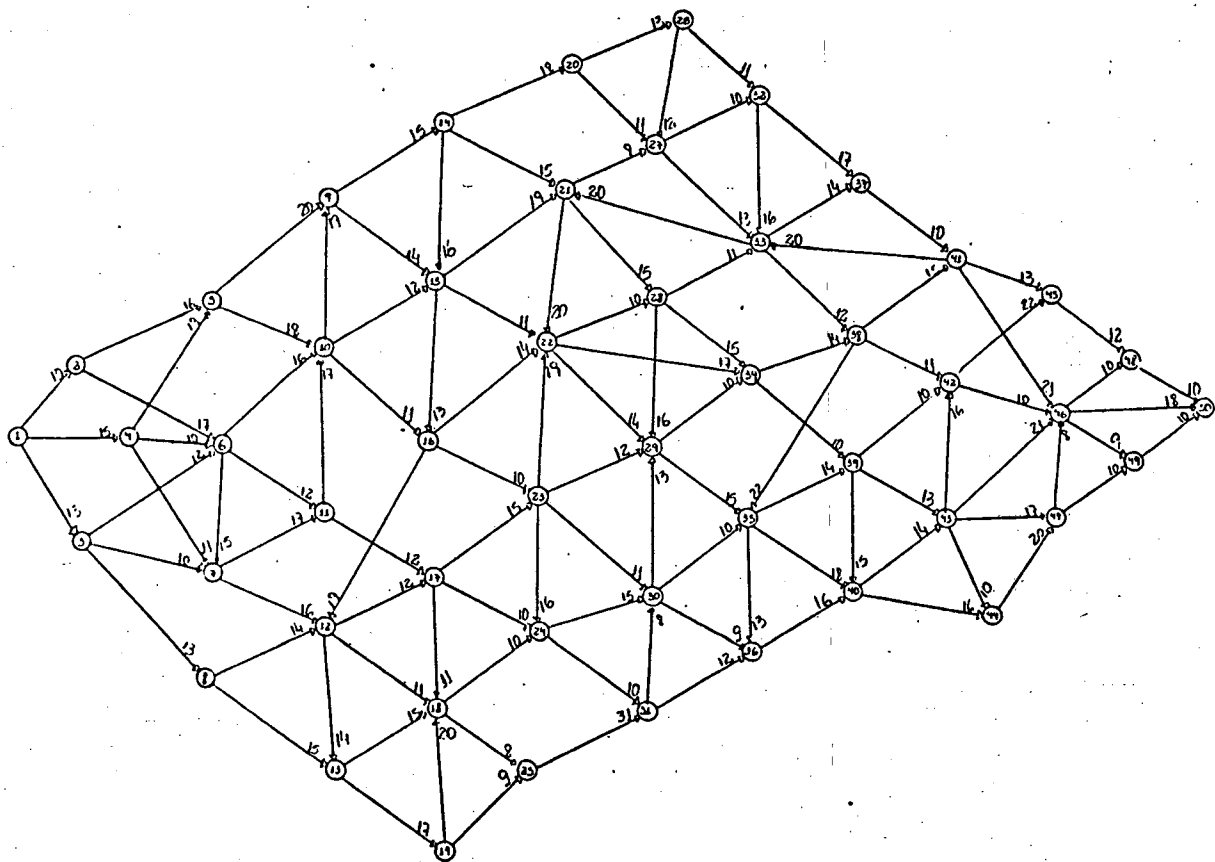
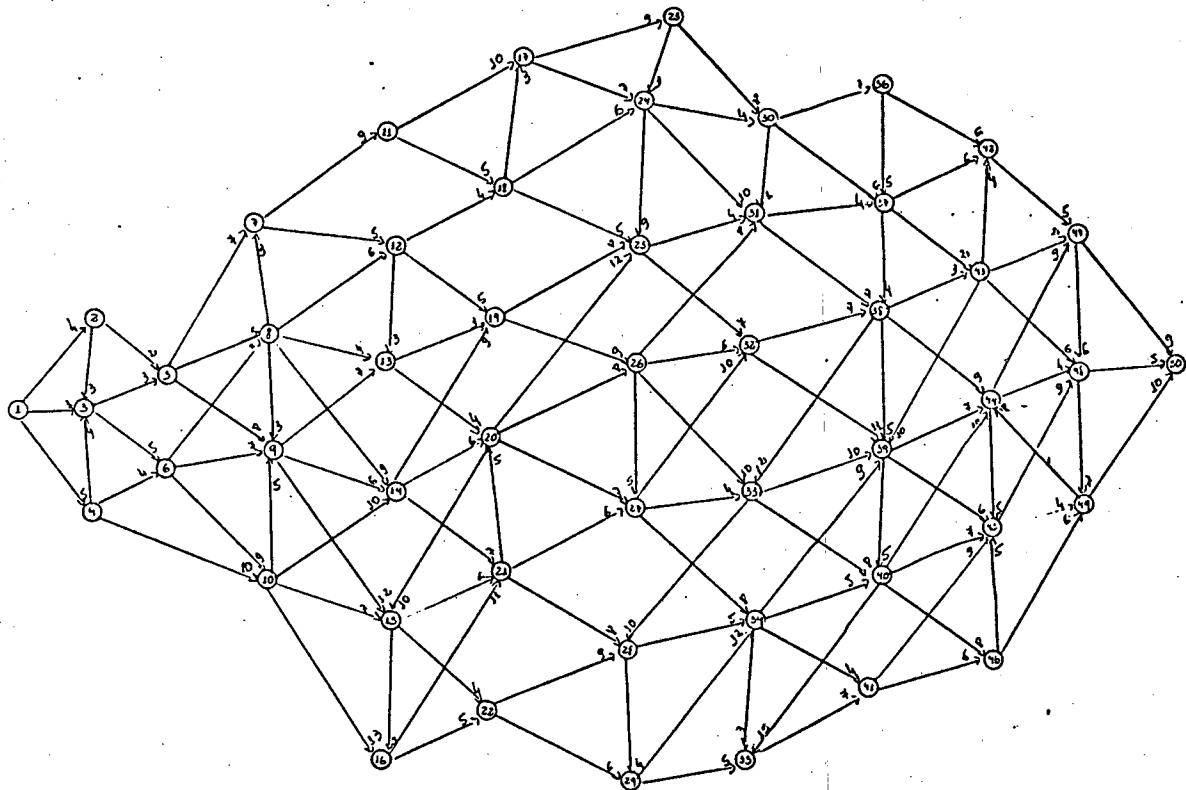


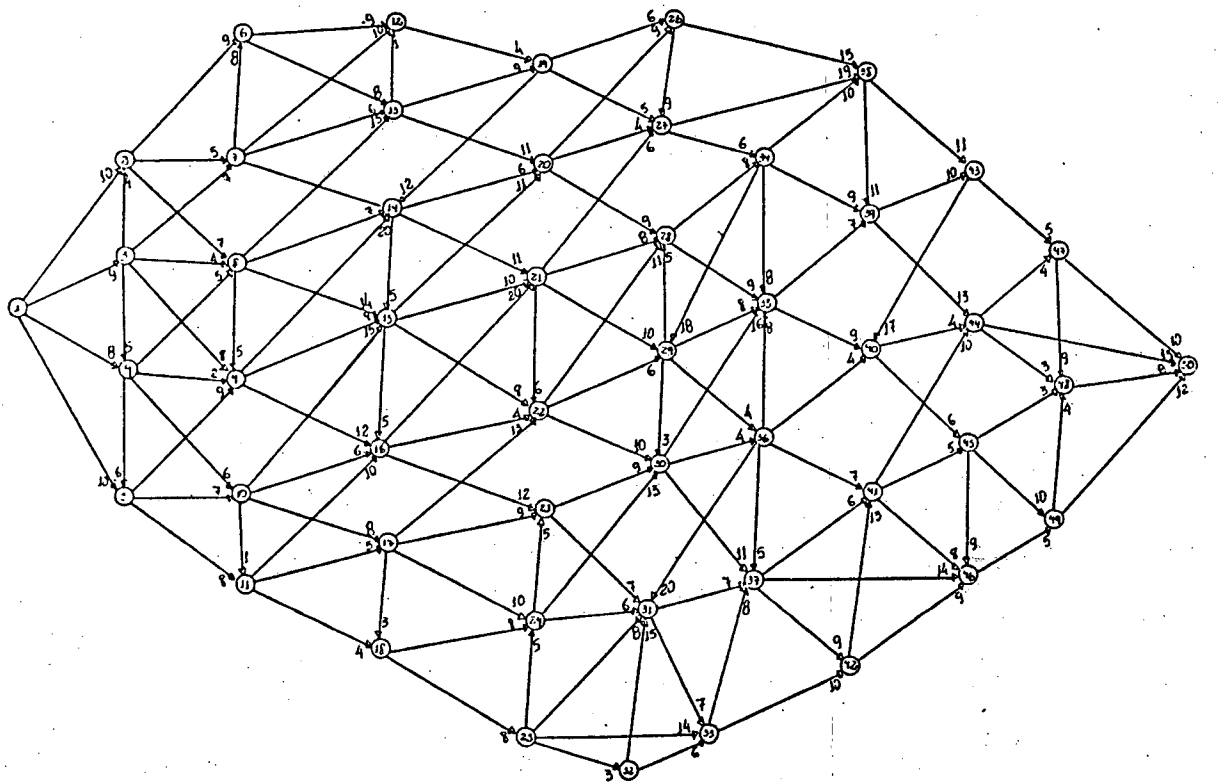
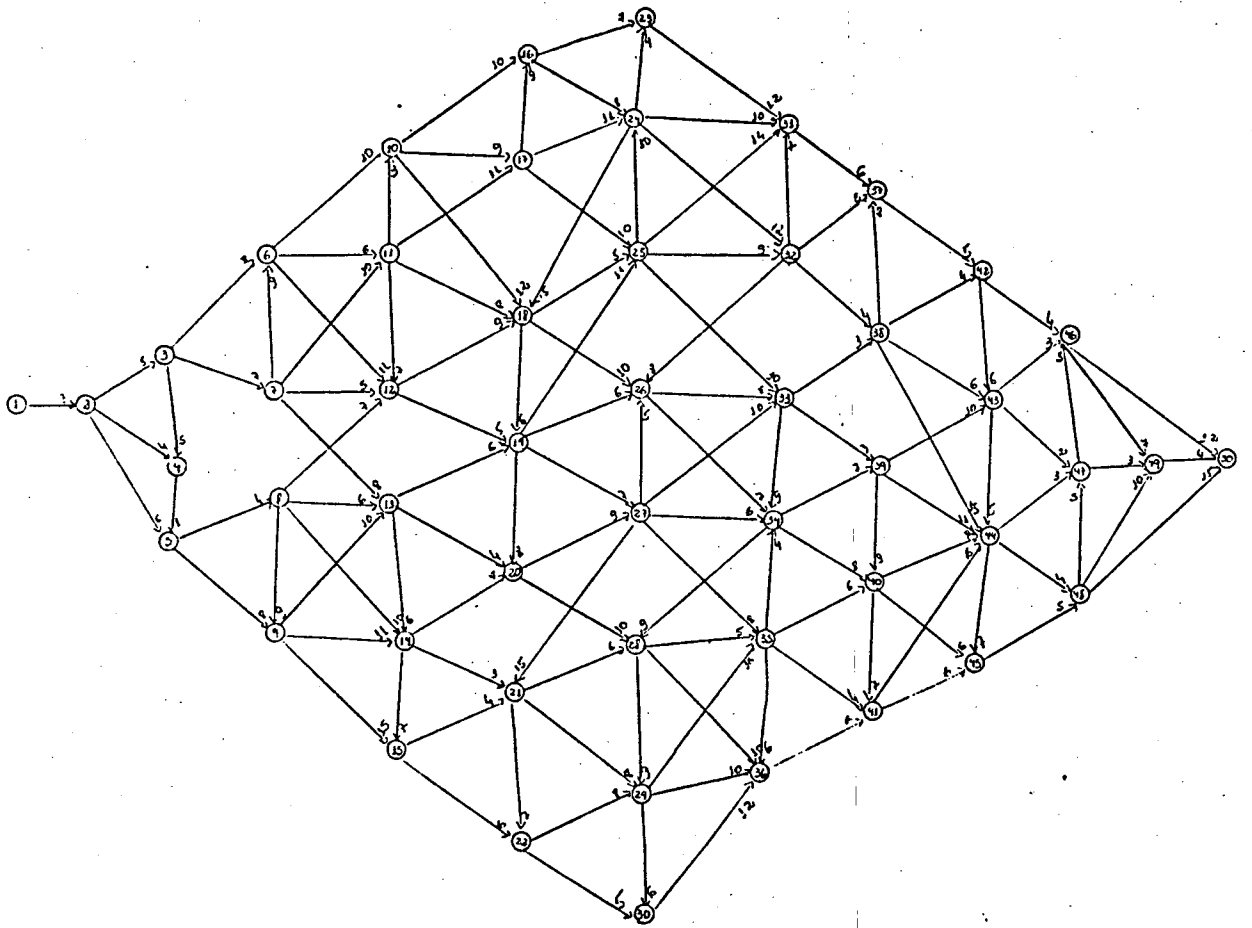


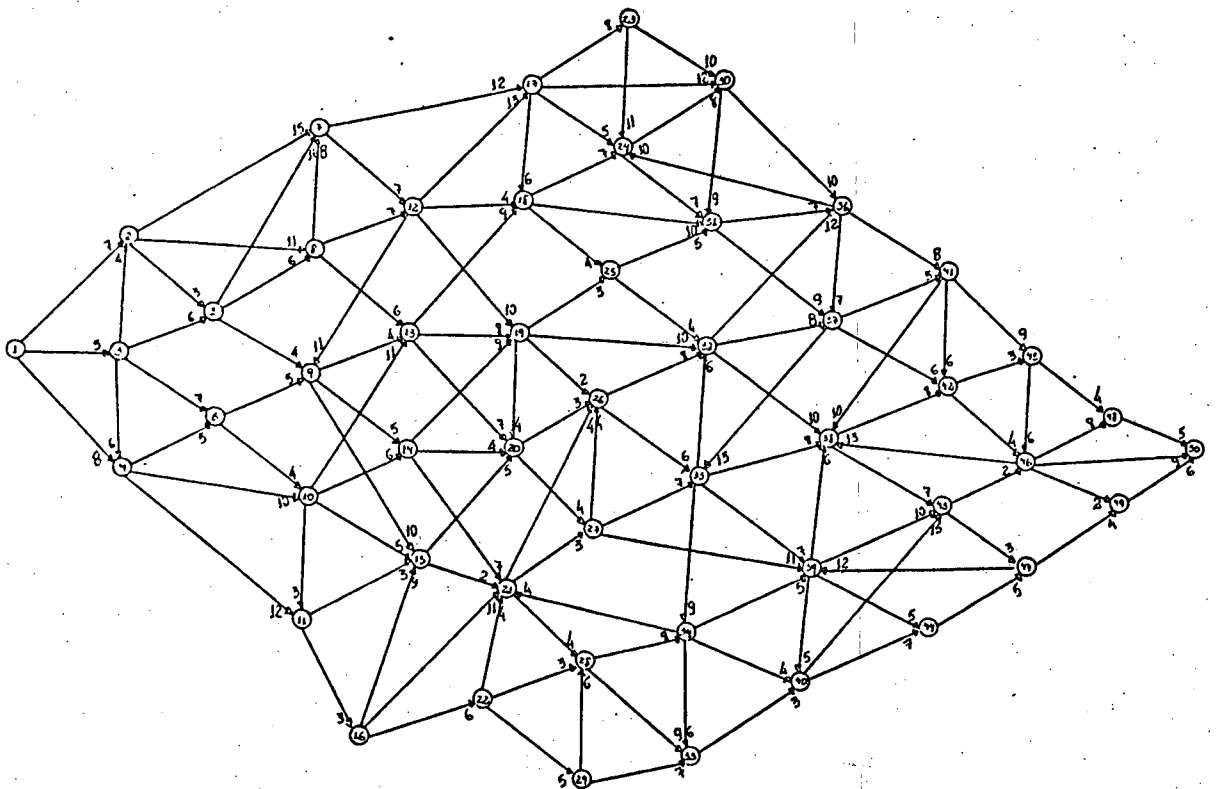
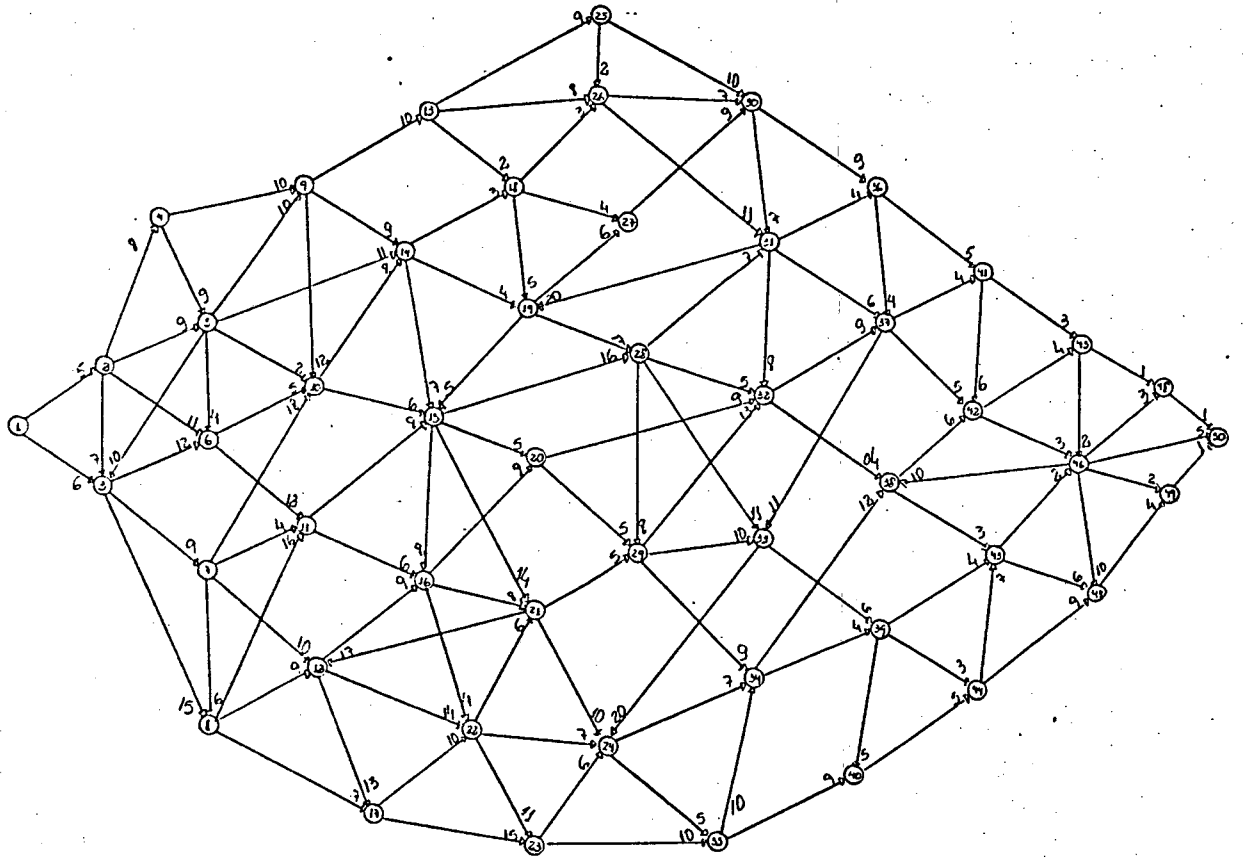


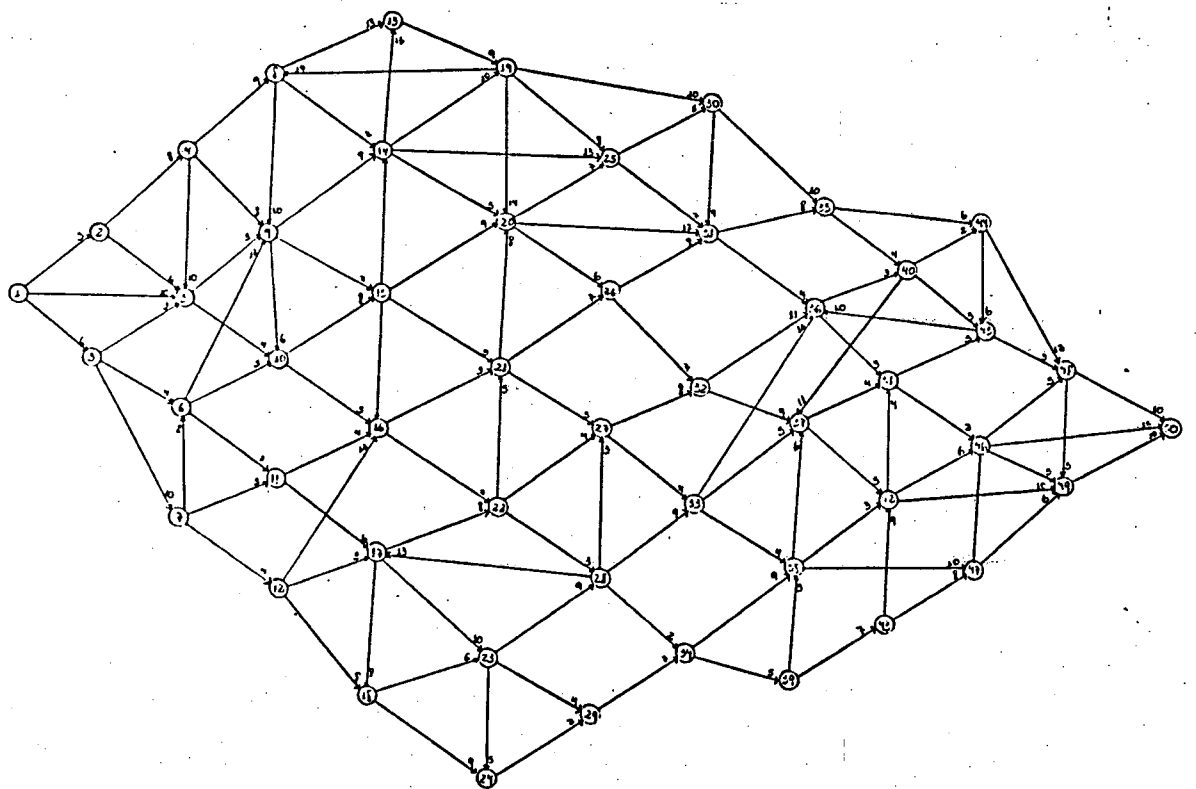
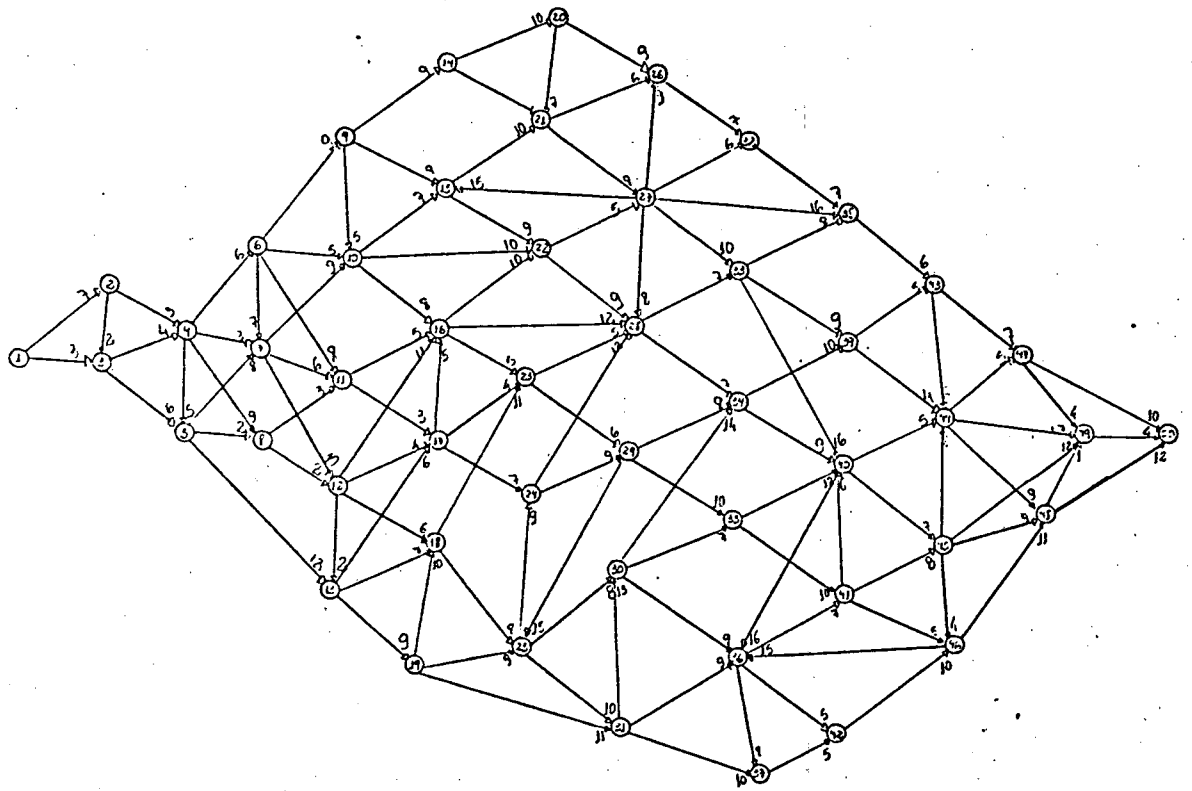


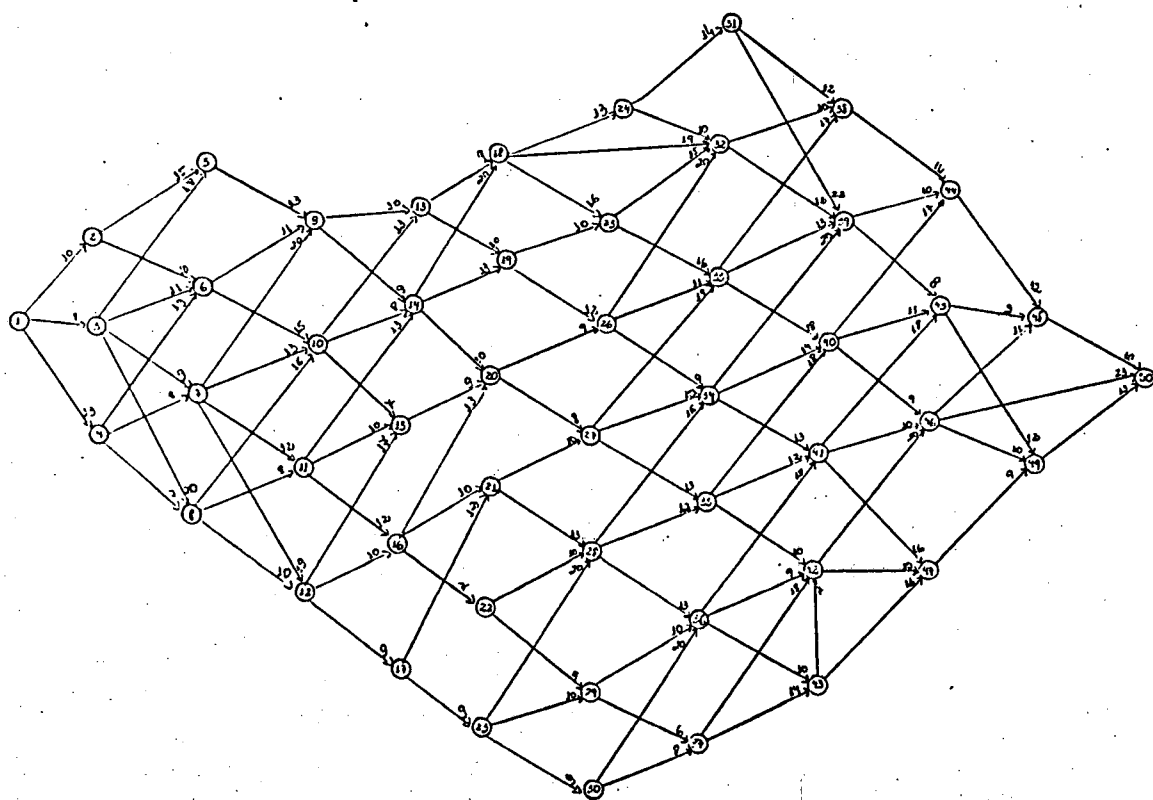
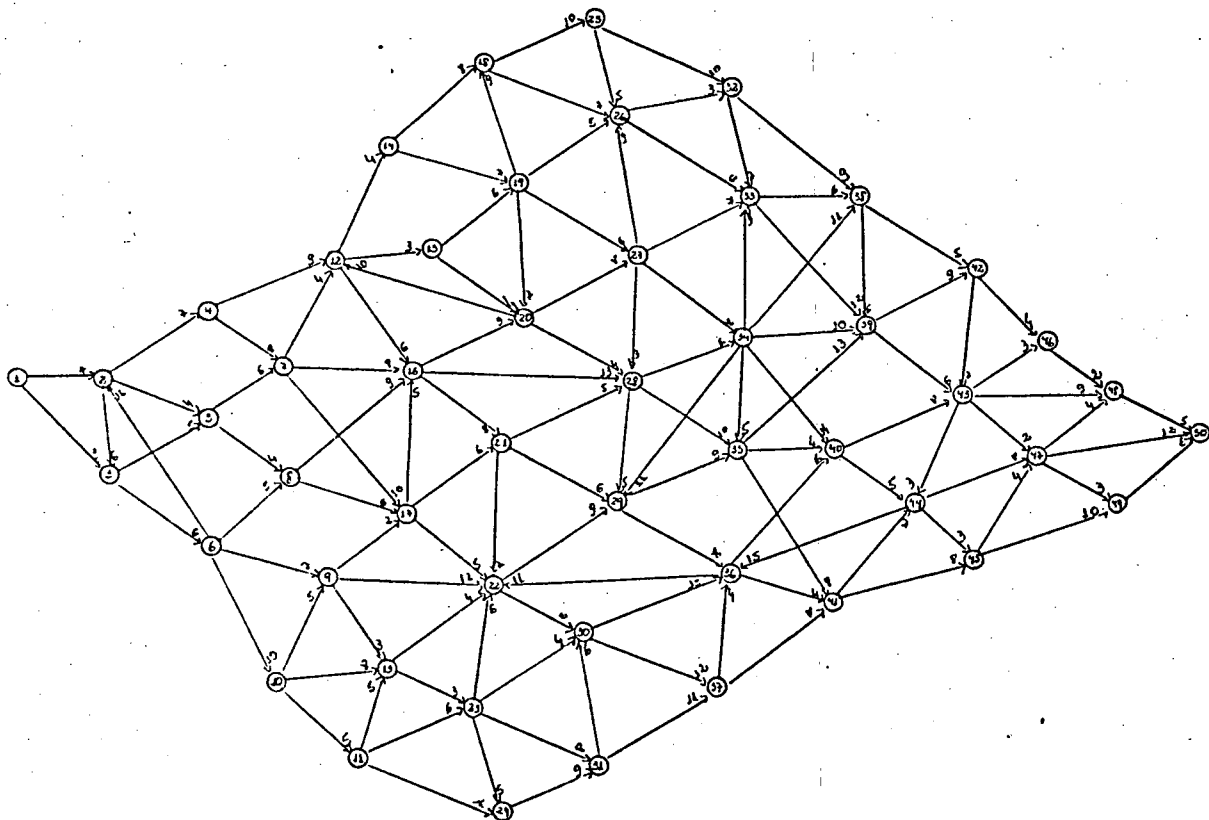






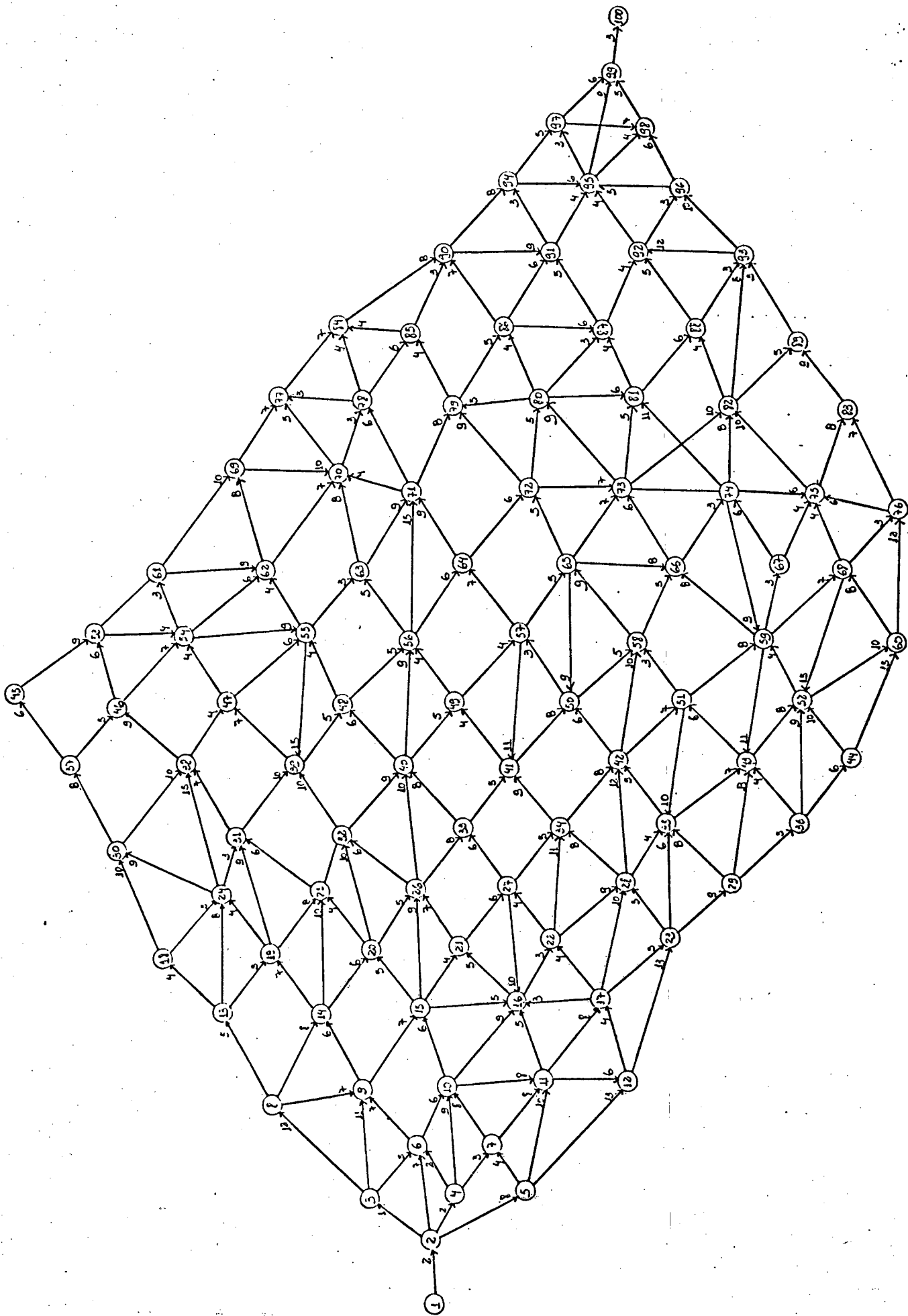


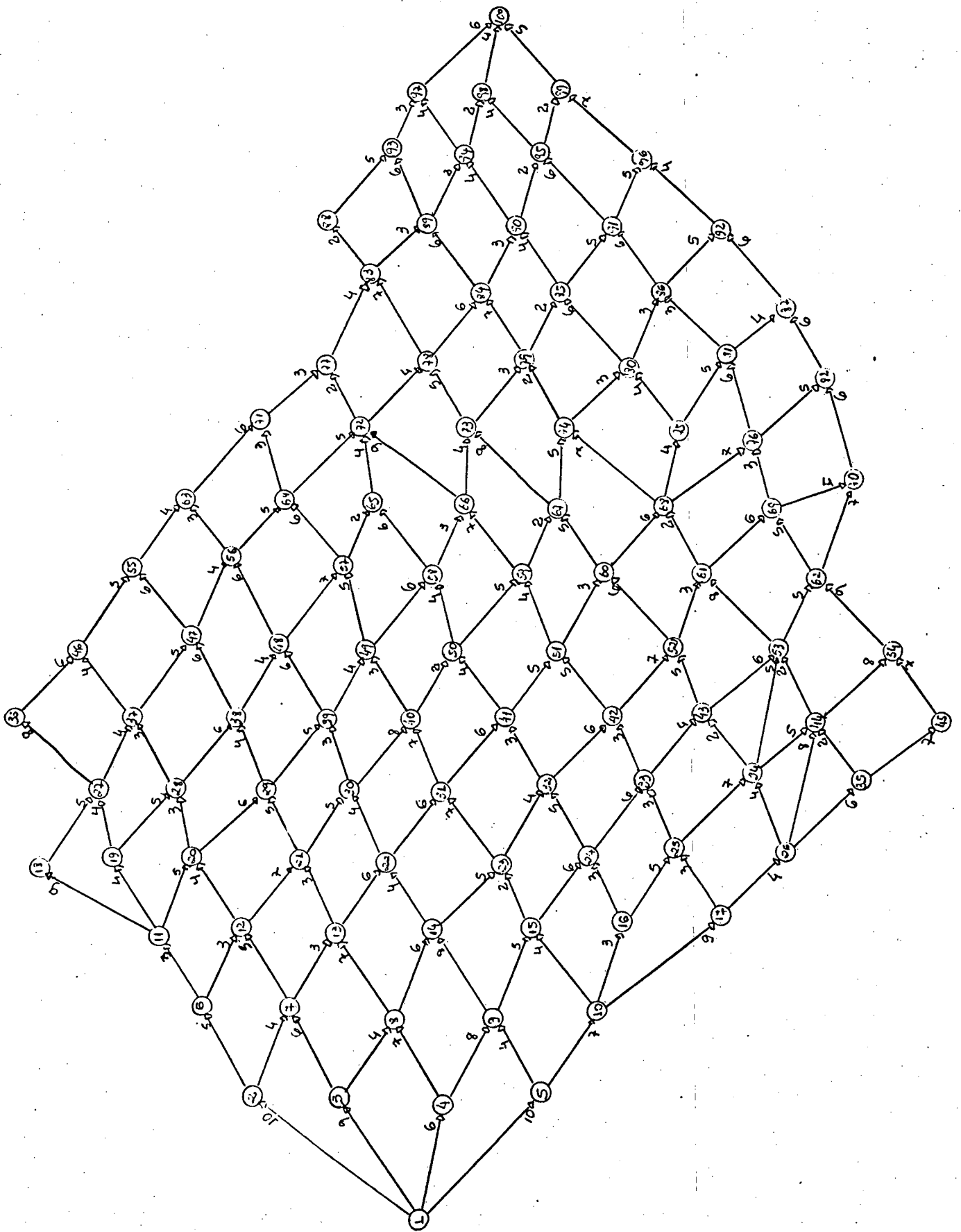


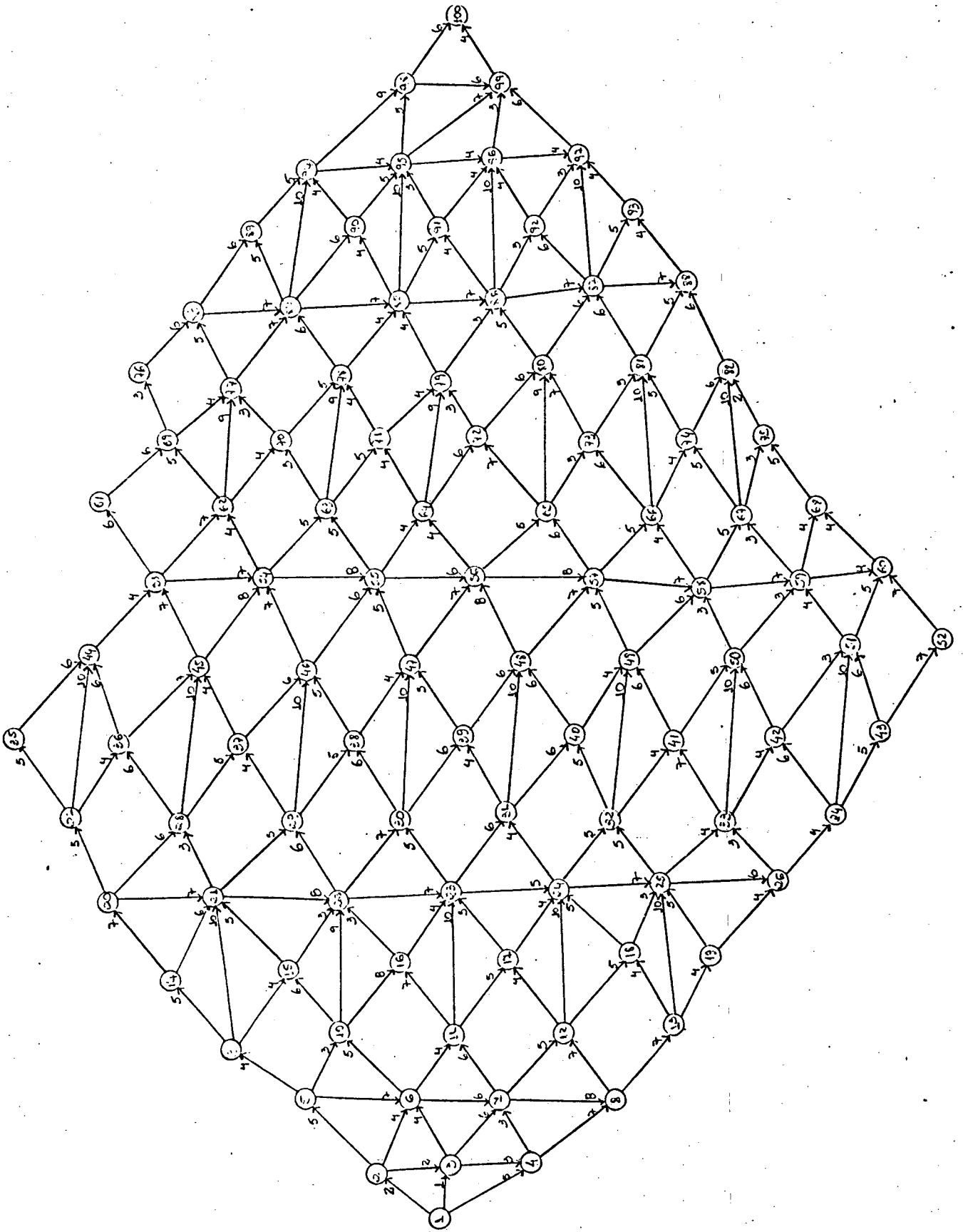


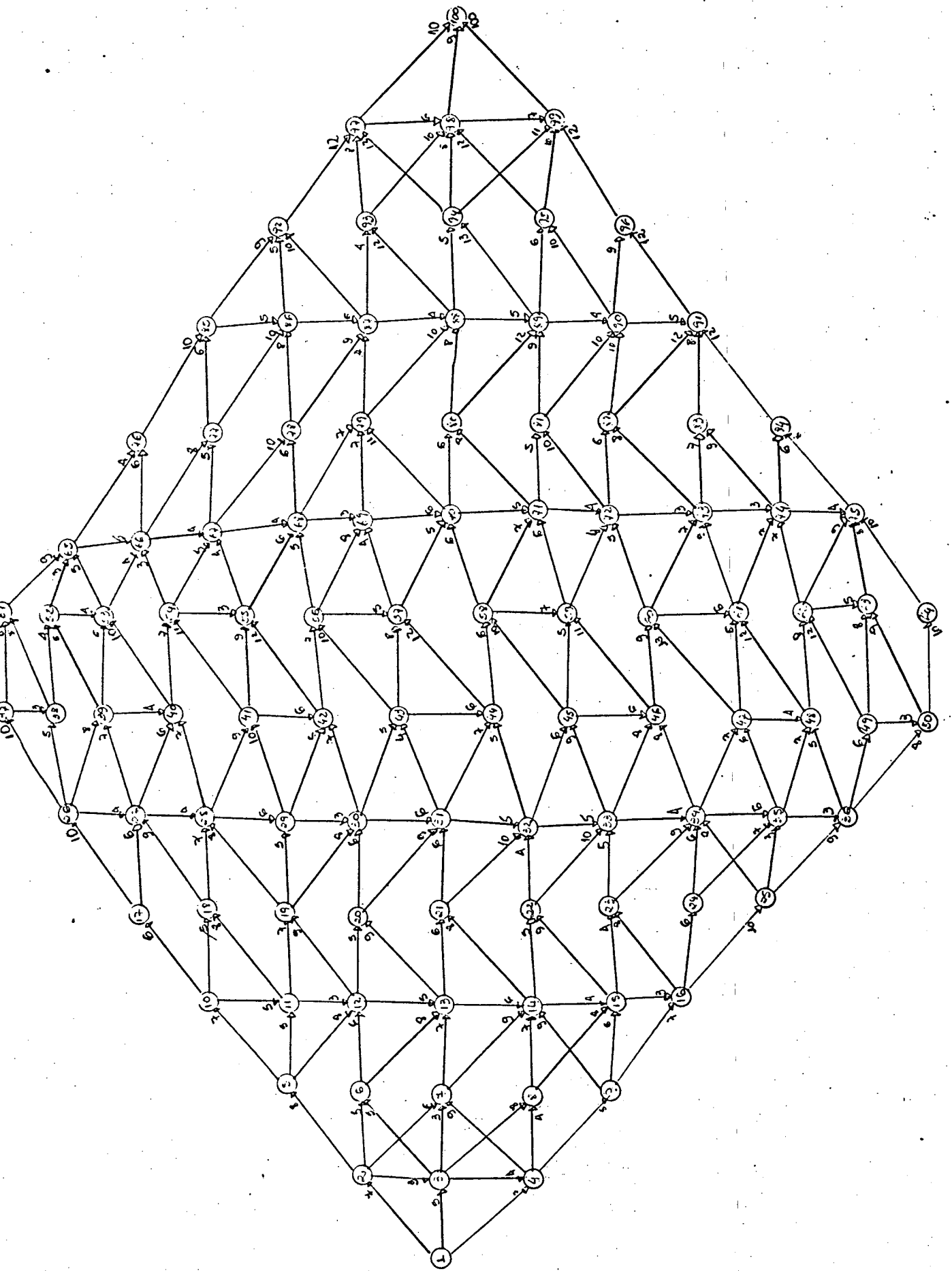
F I G U R A 15

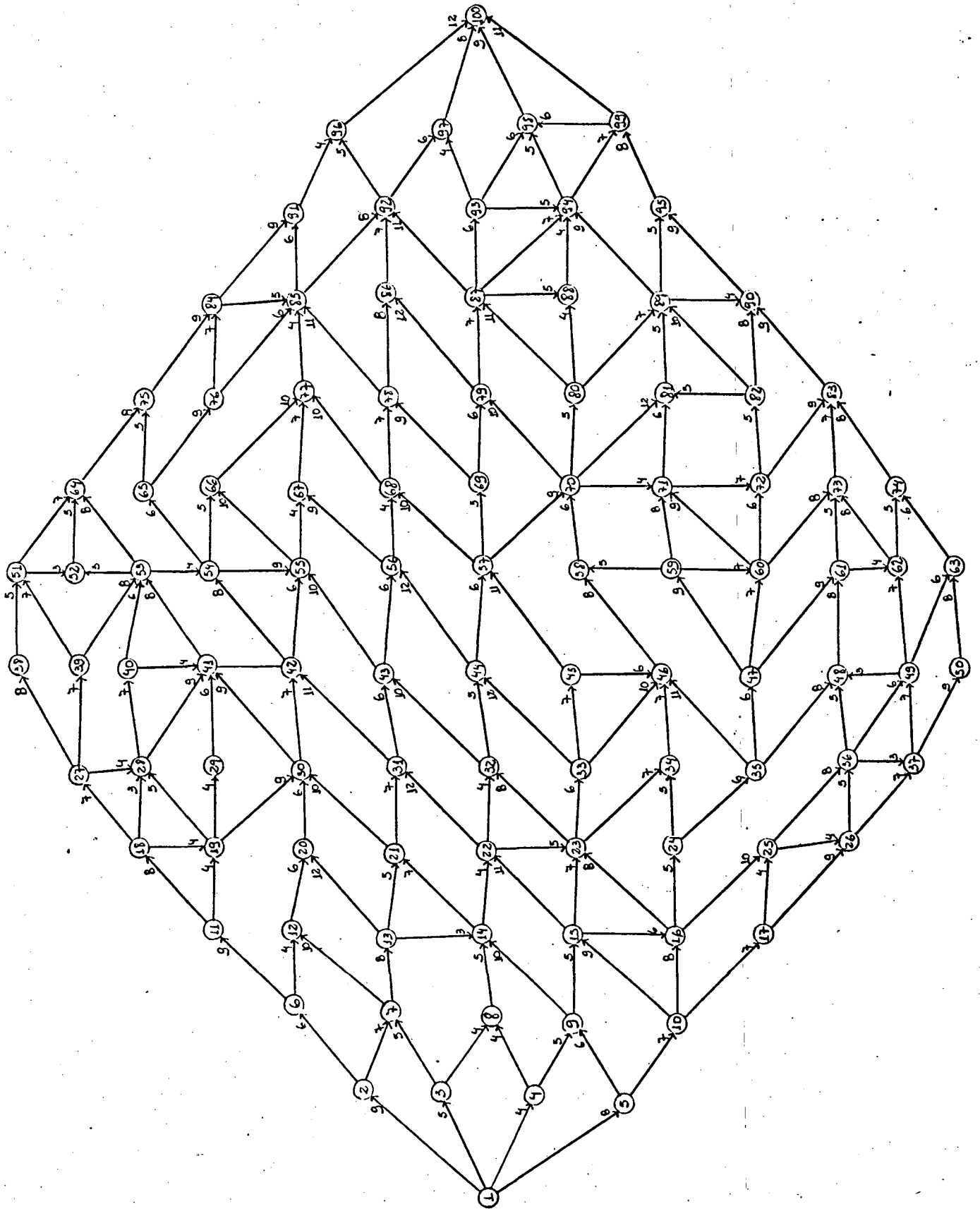
GRAFOS COM 100 VÉRTICES

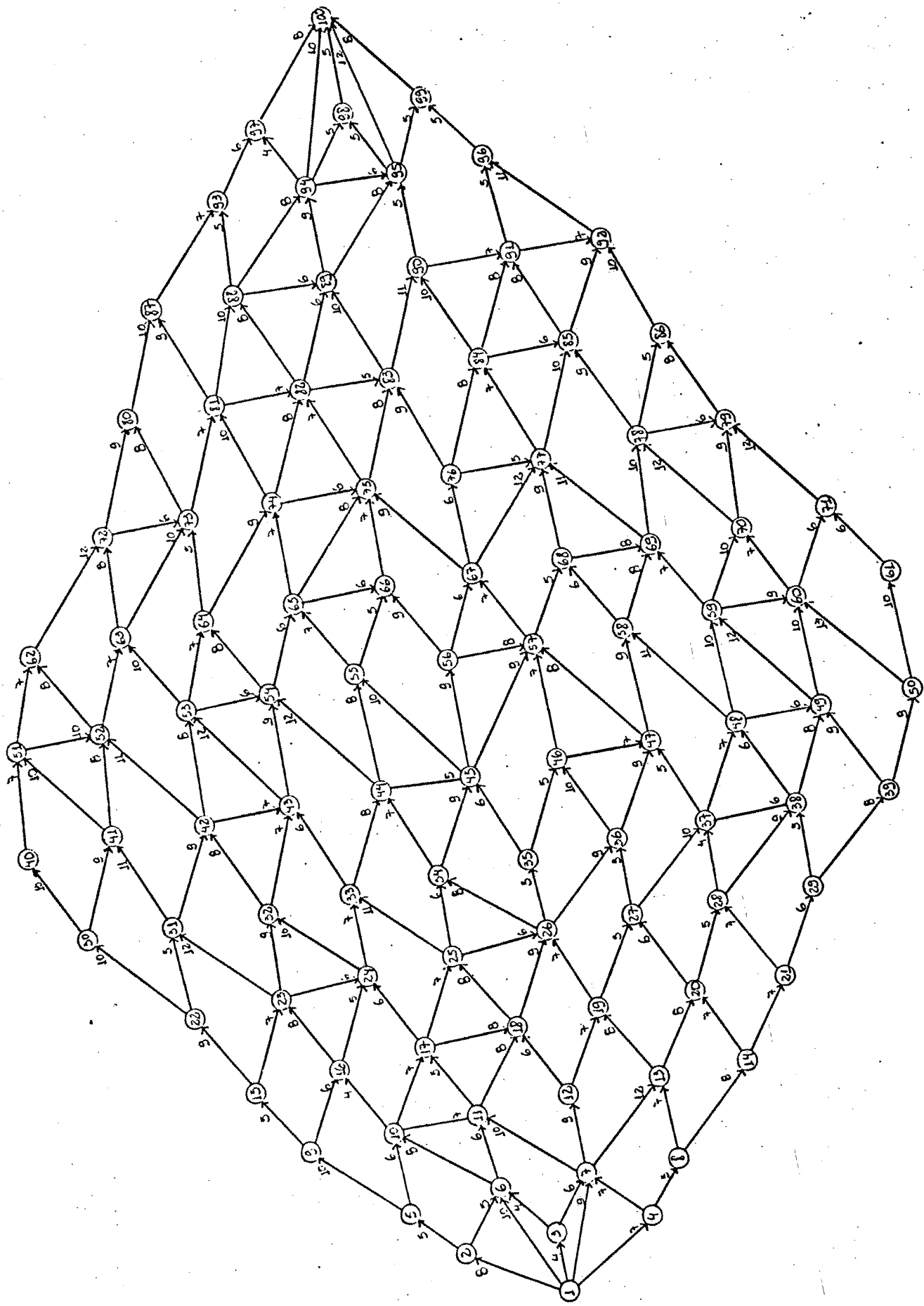


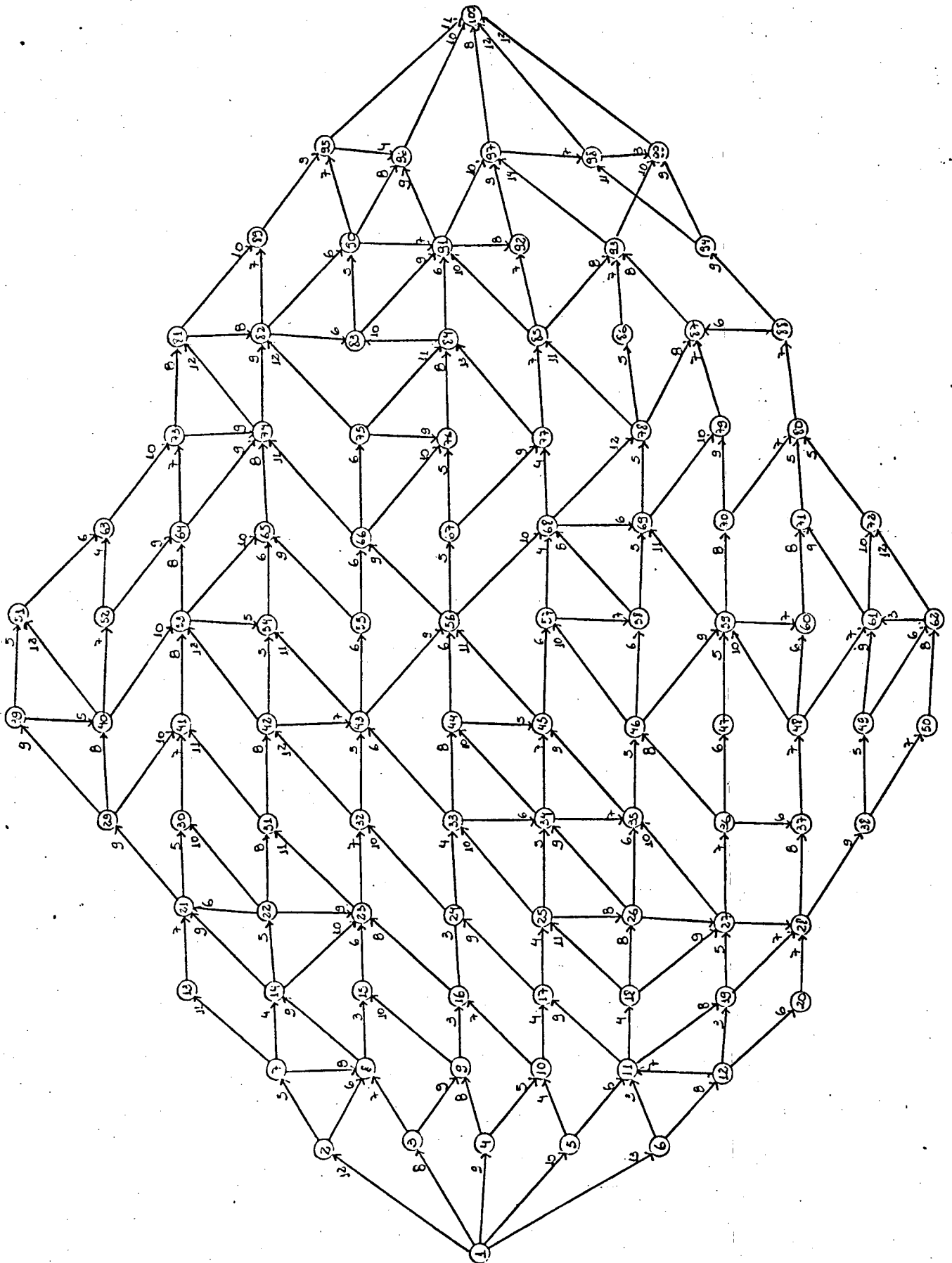


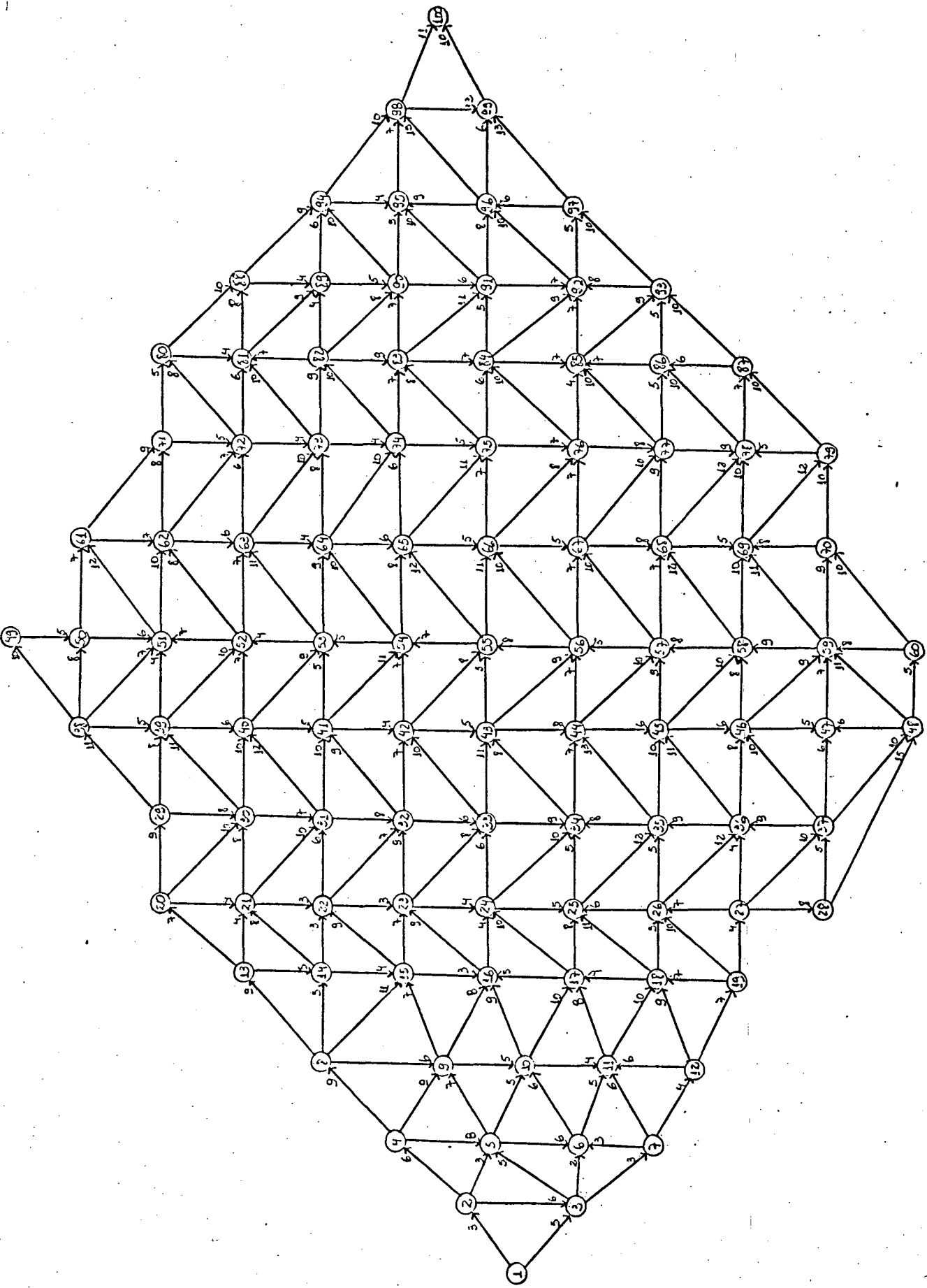


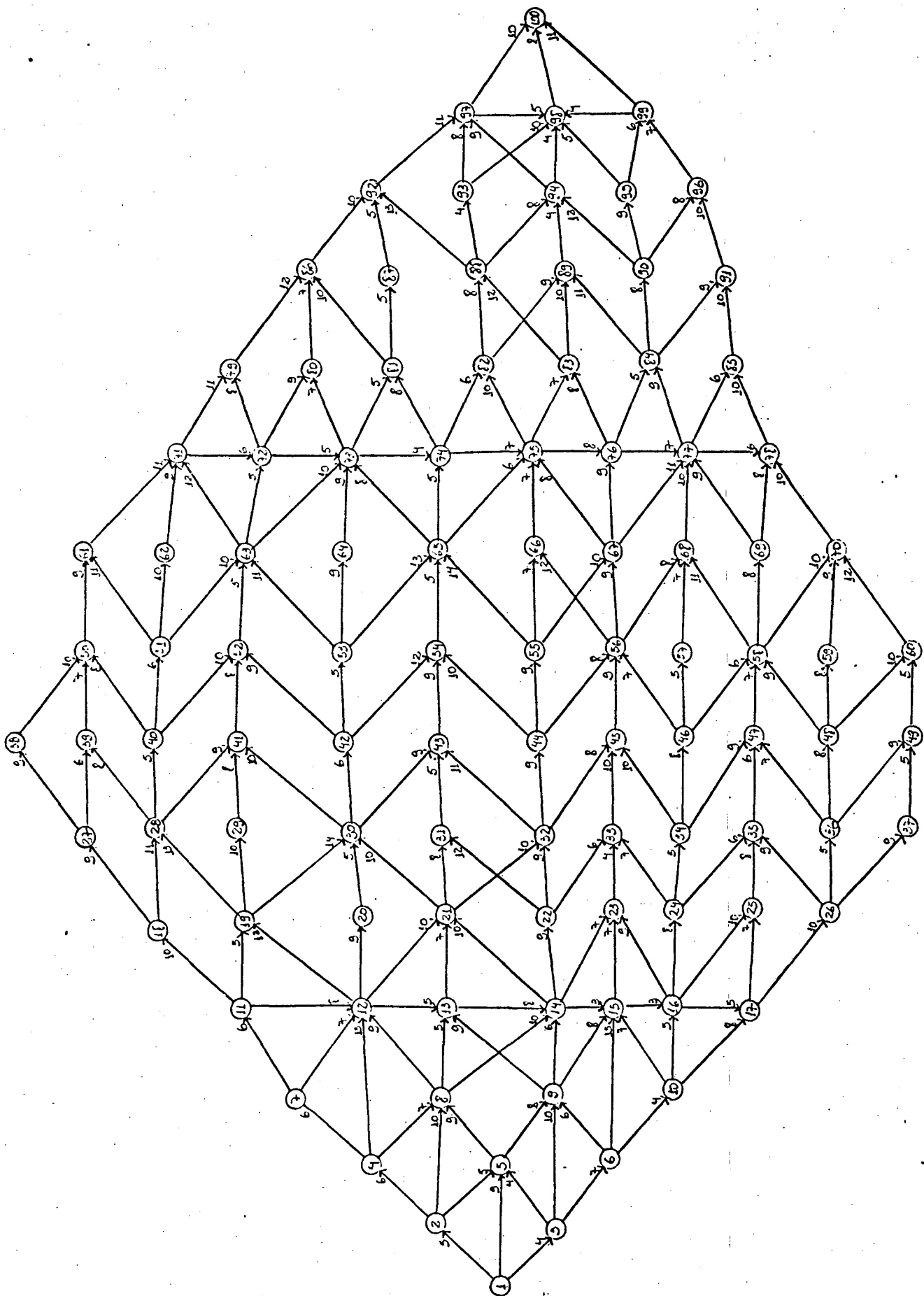


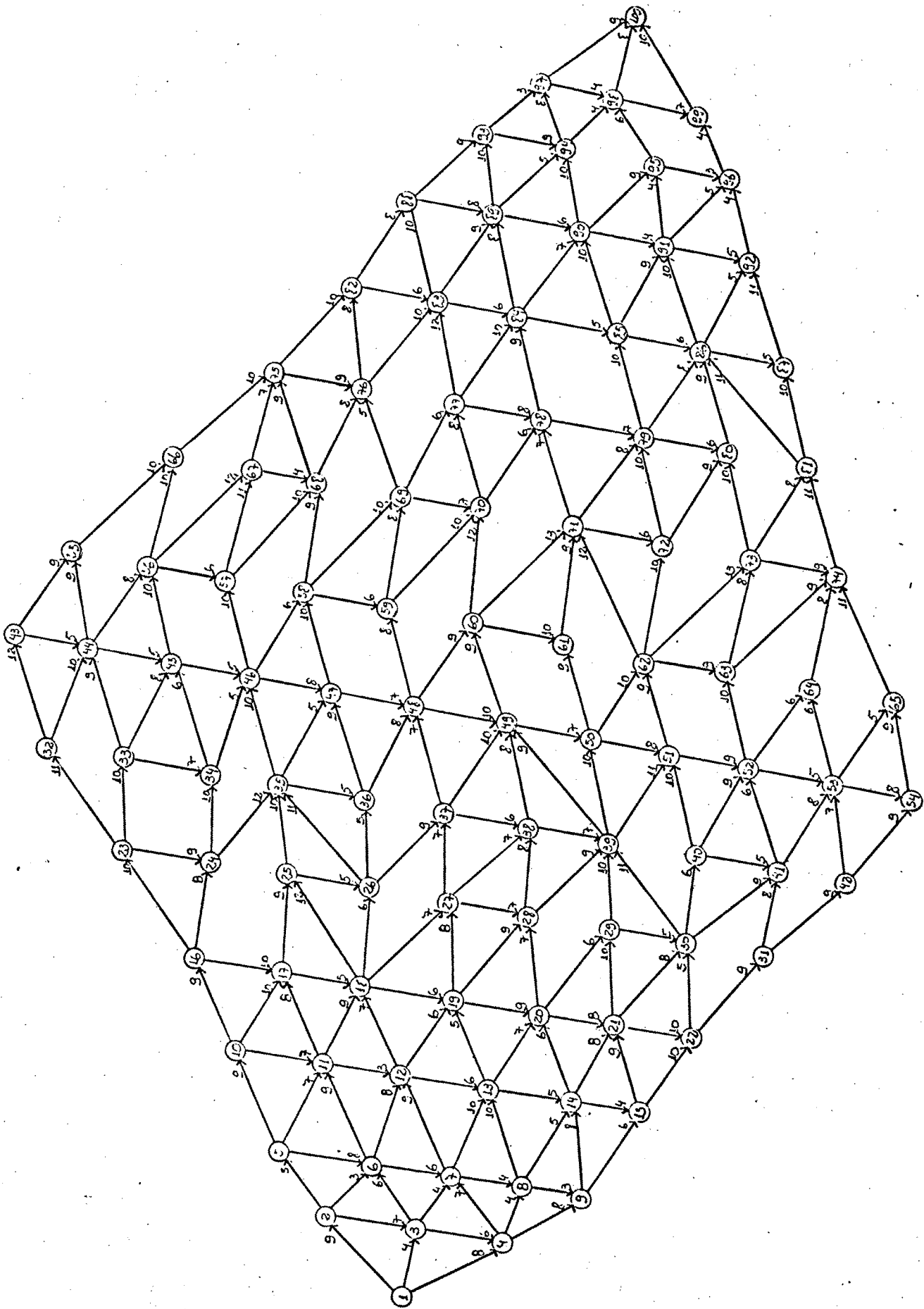


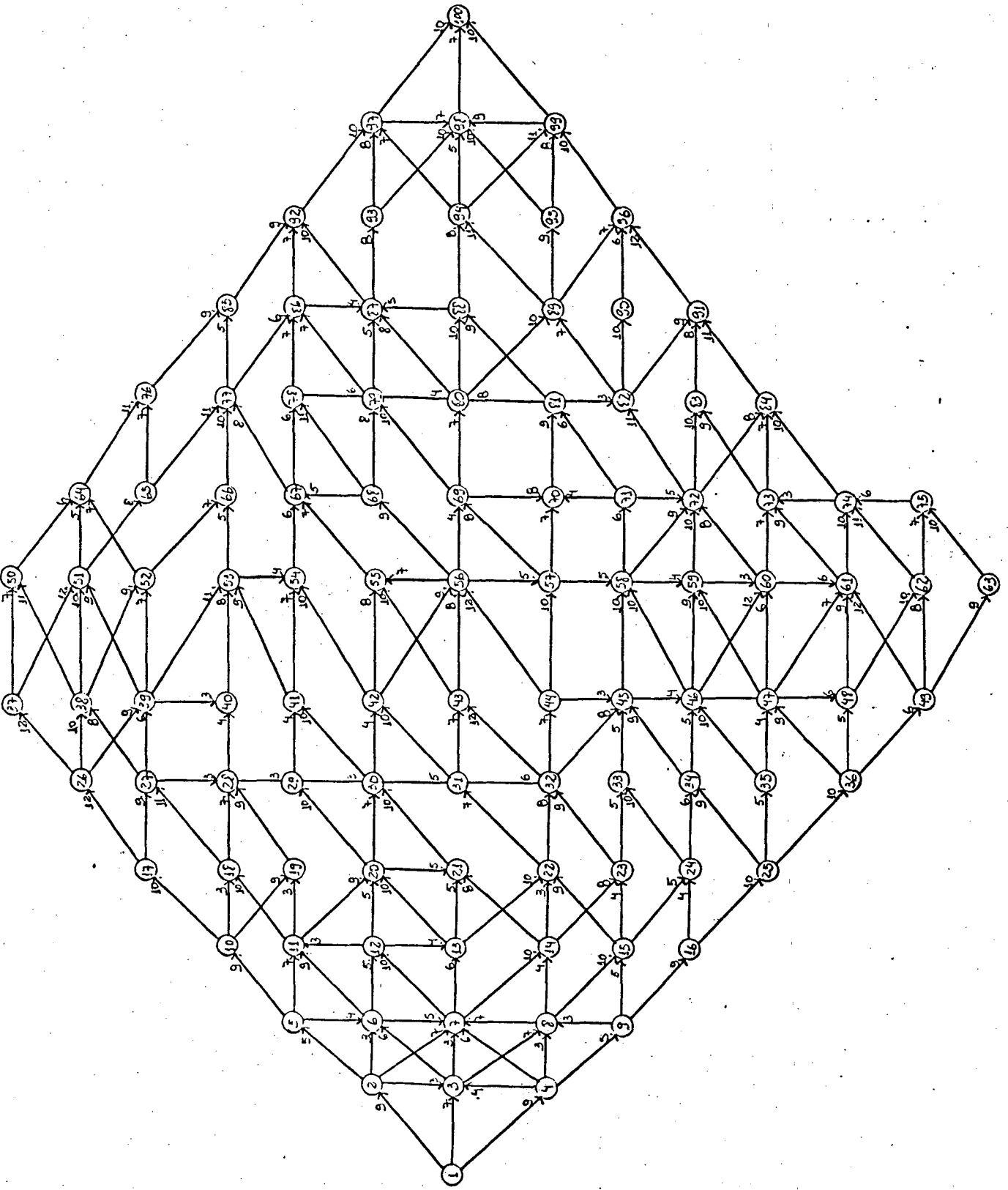


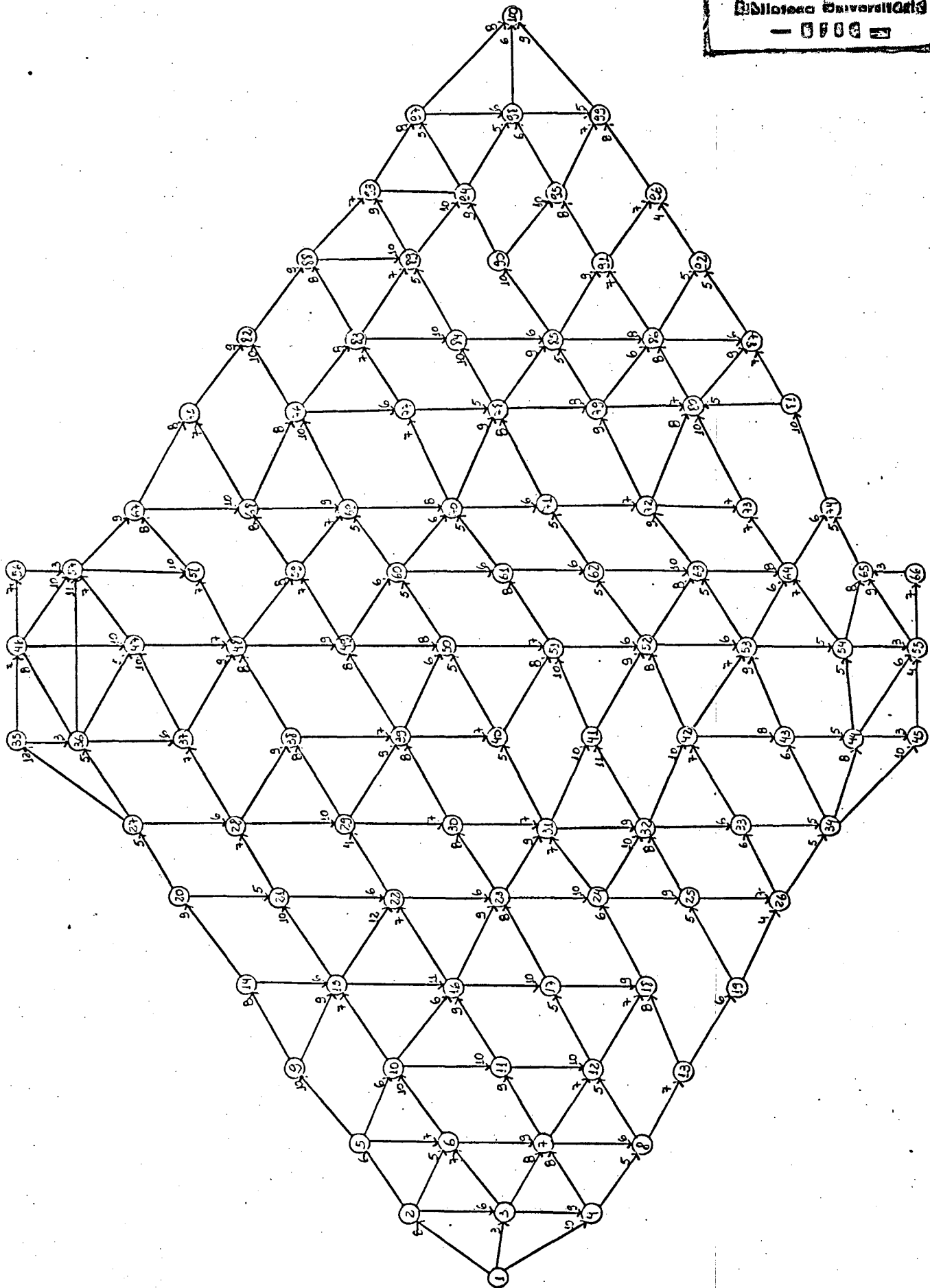


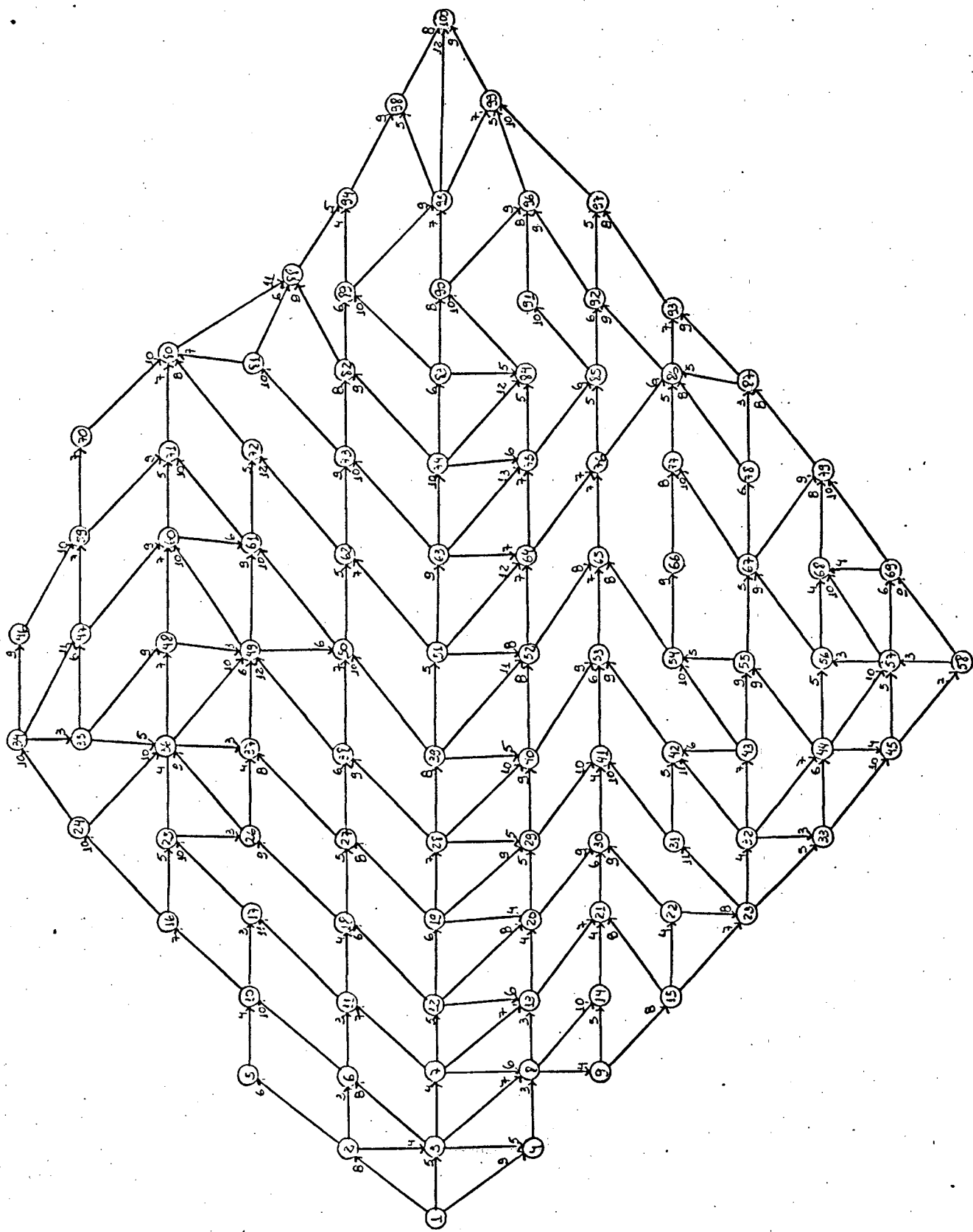


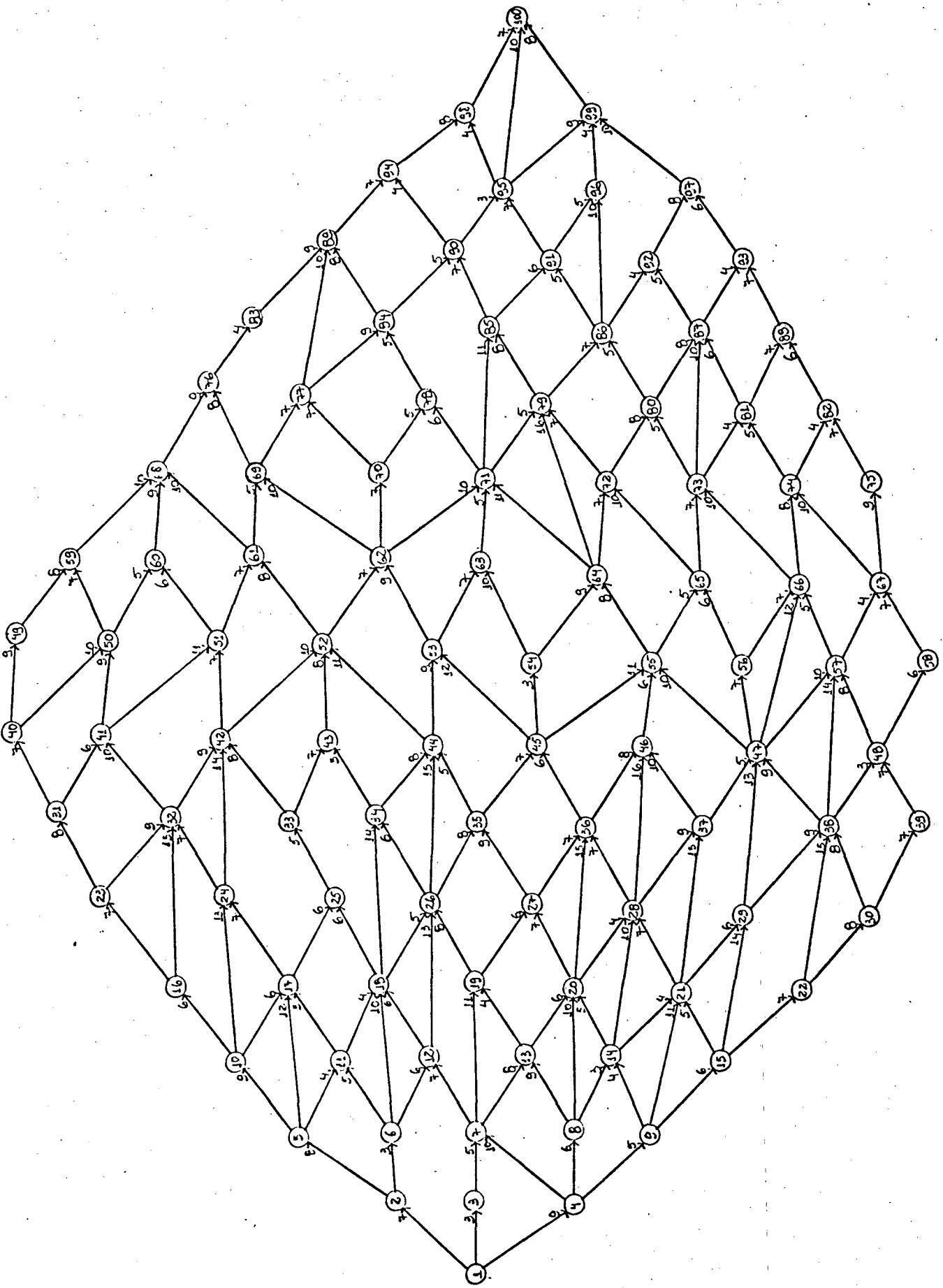


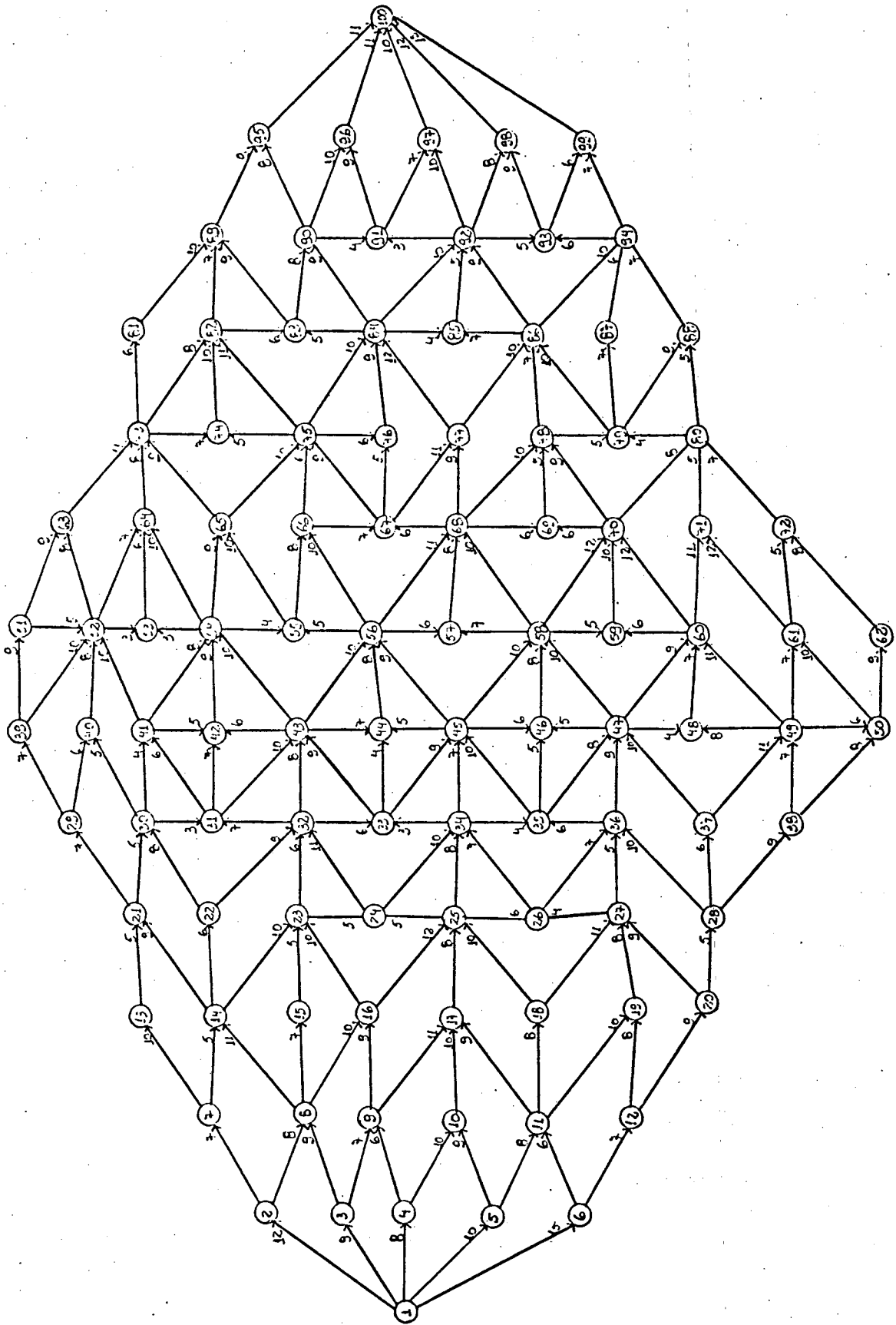


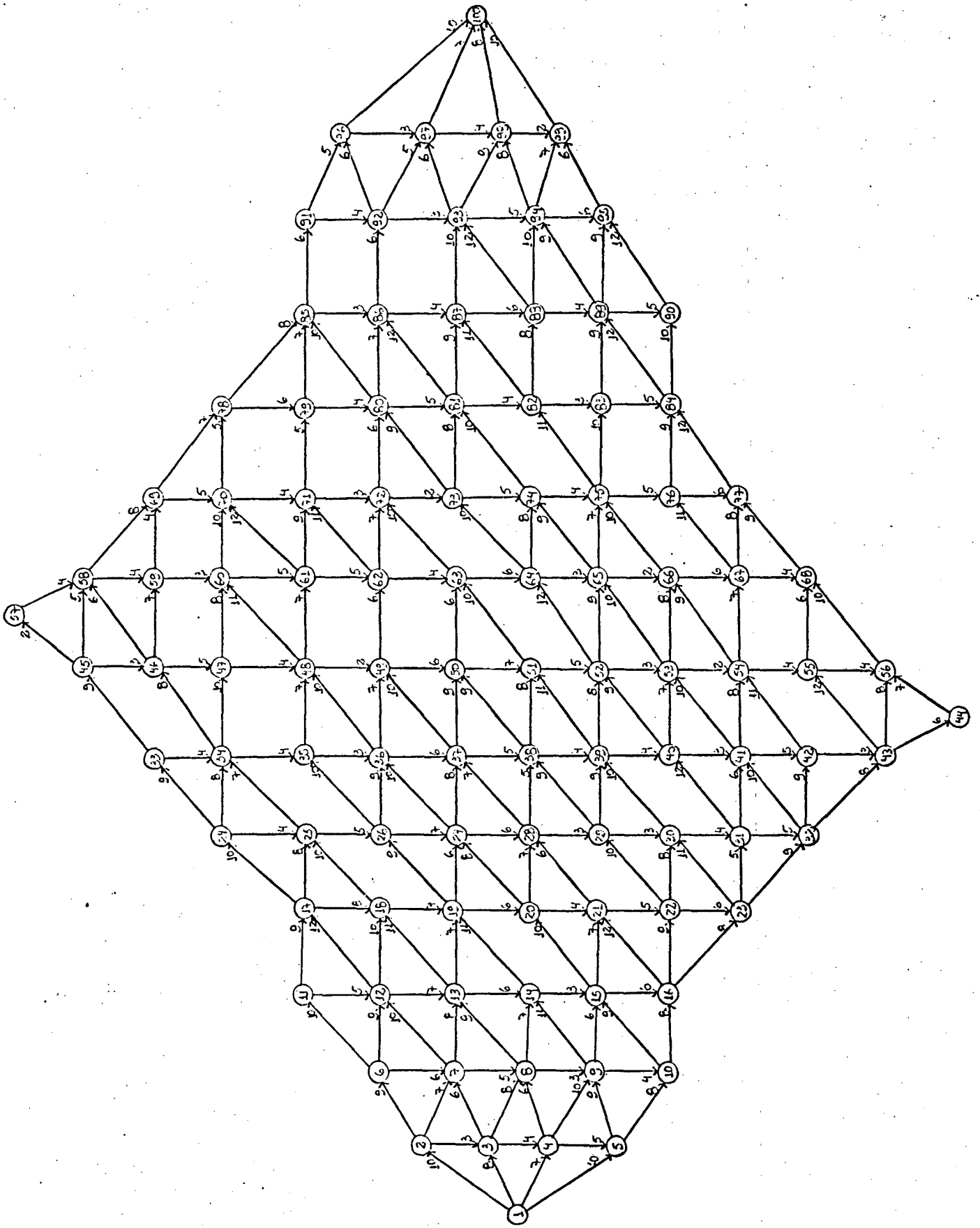


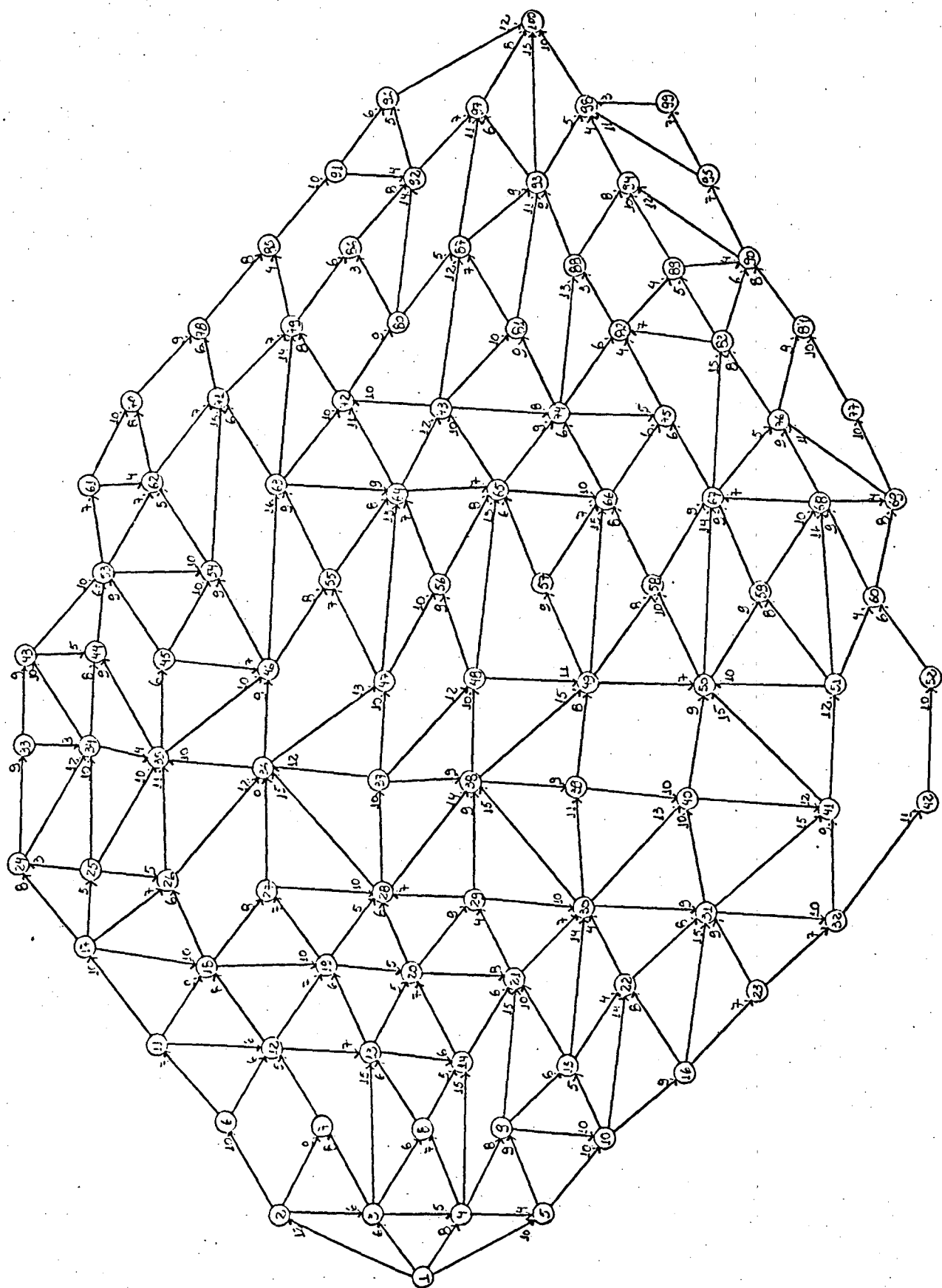


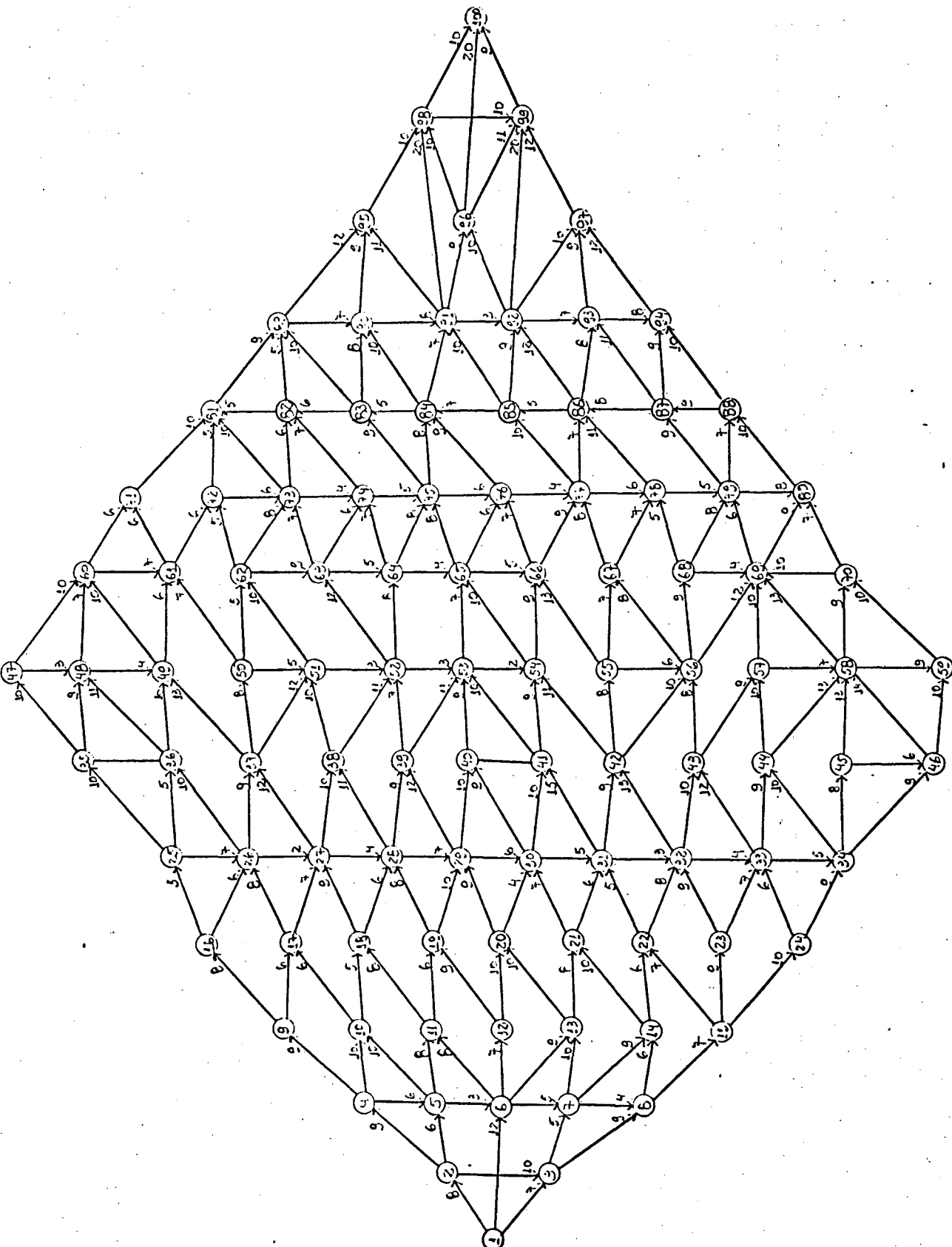


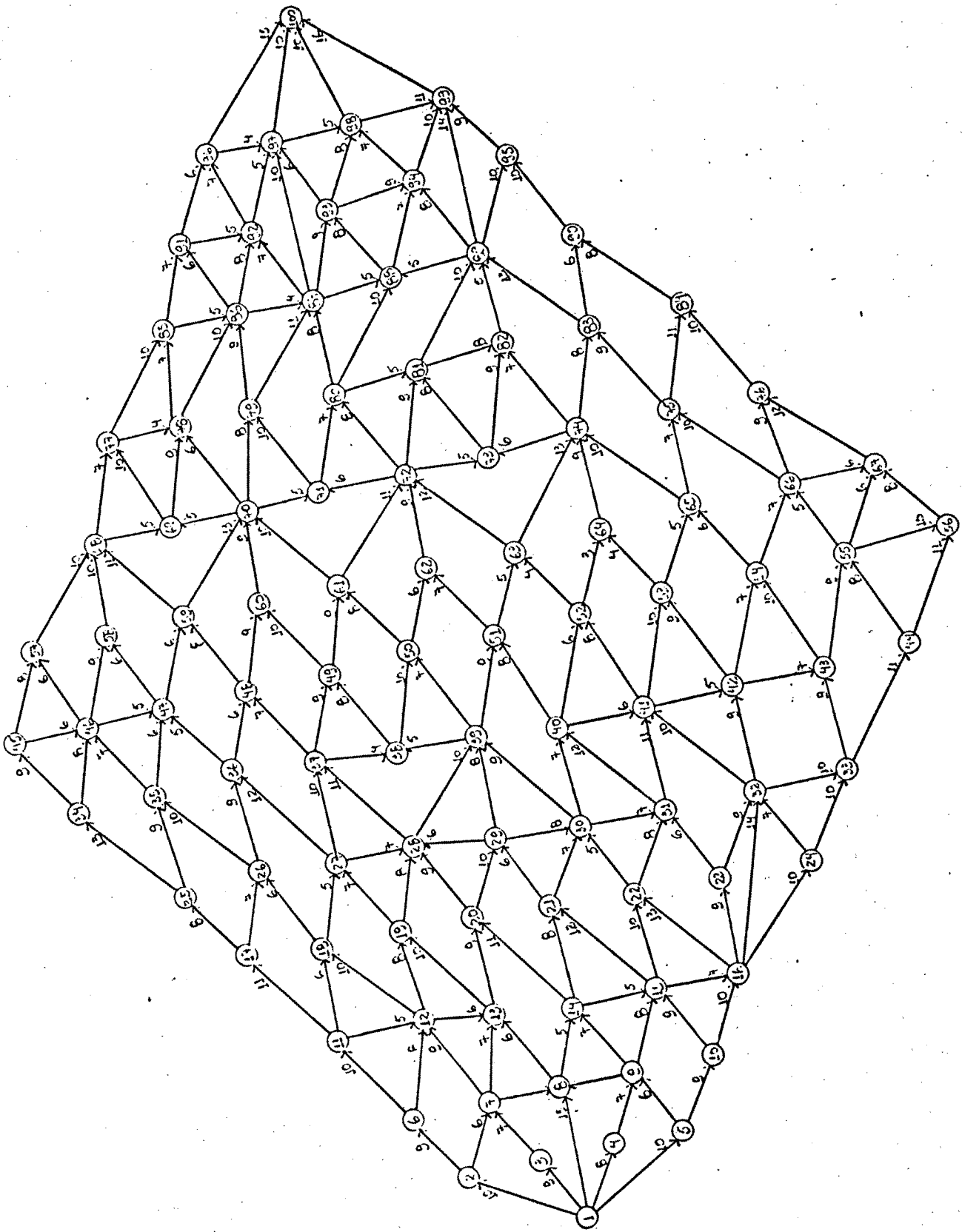


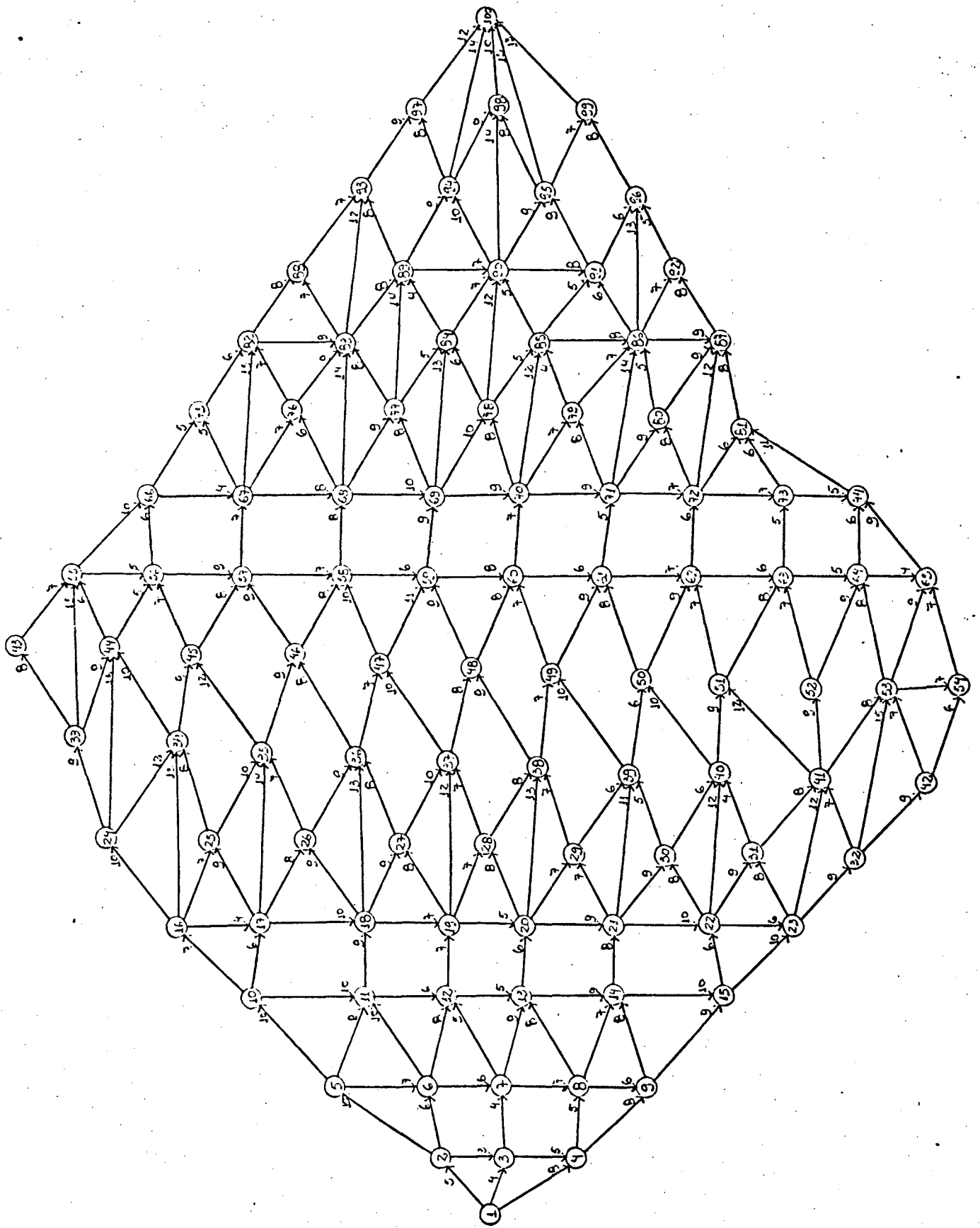






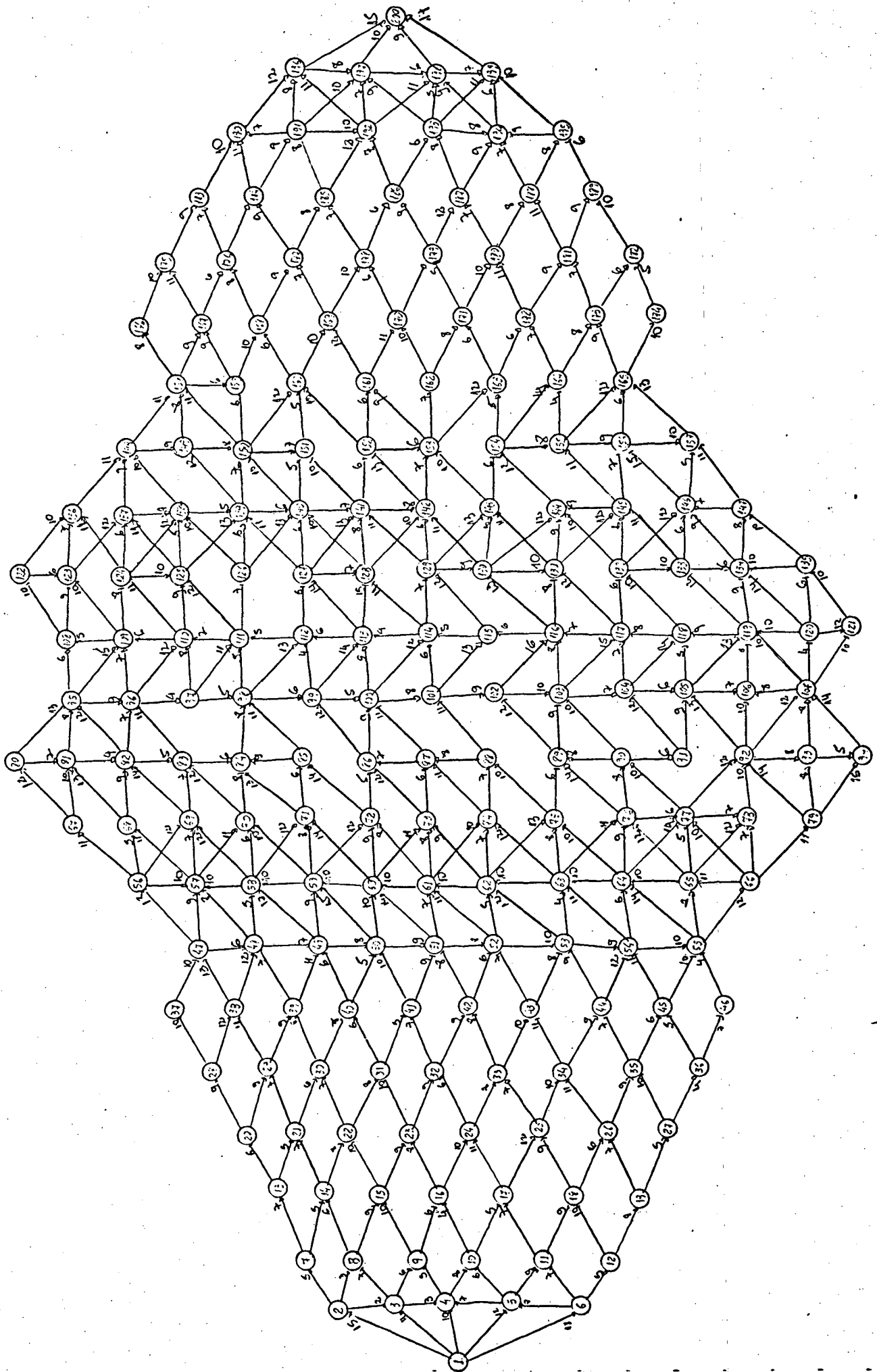


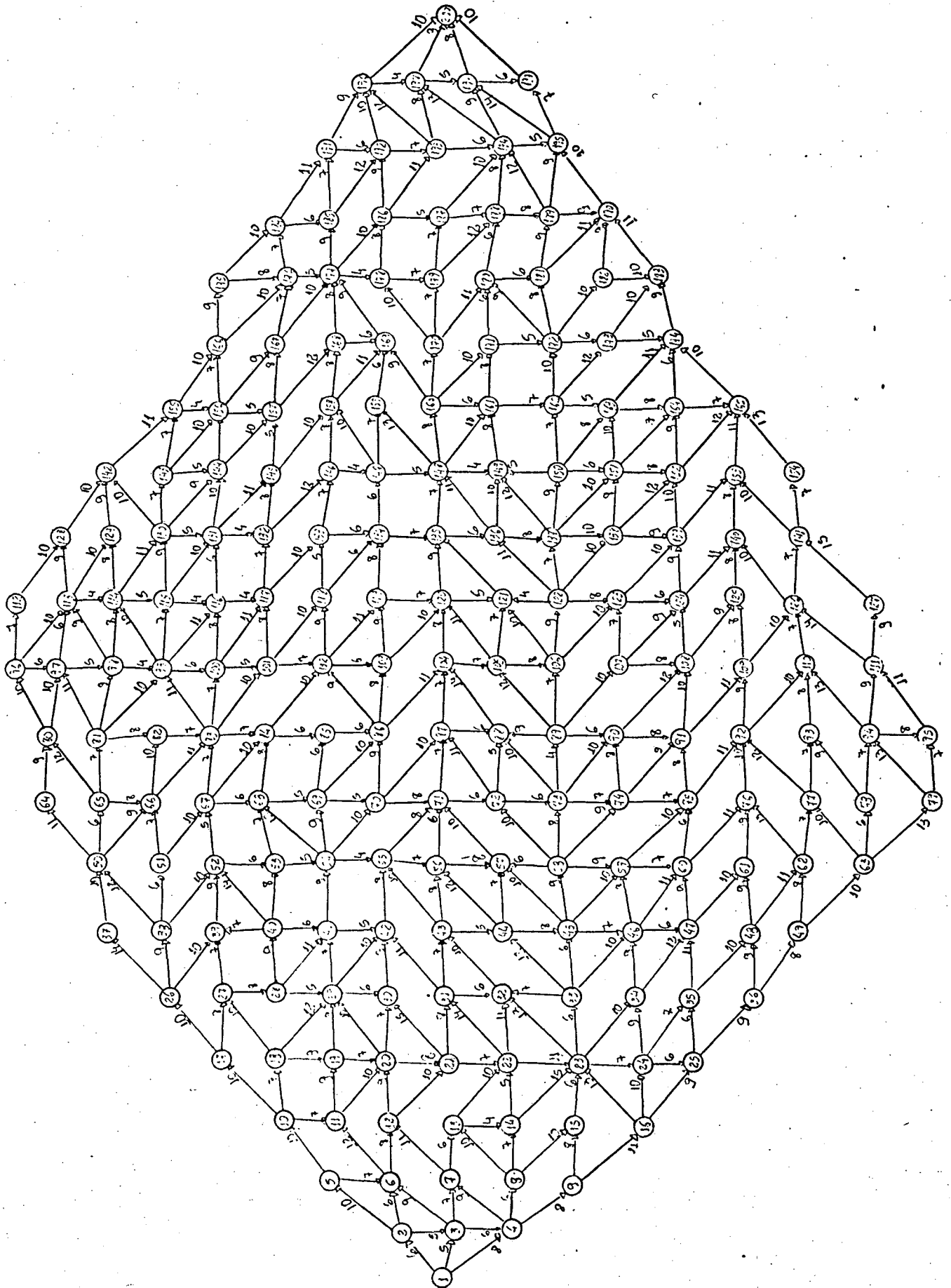


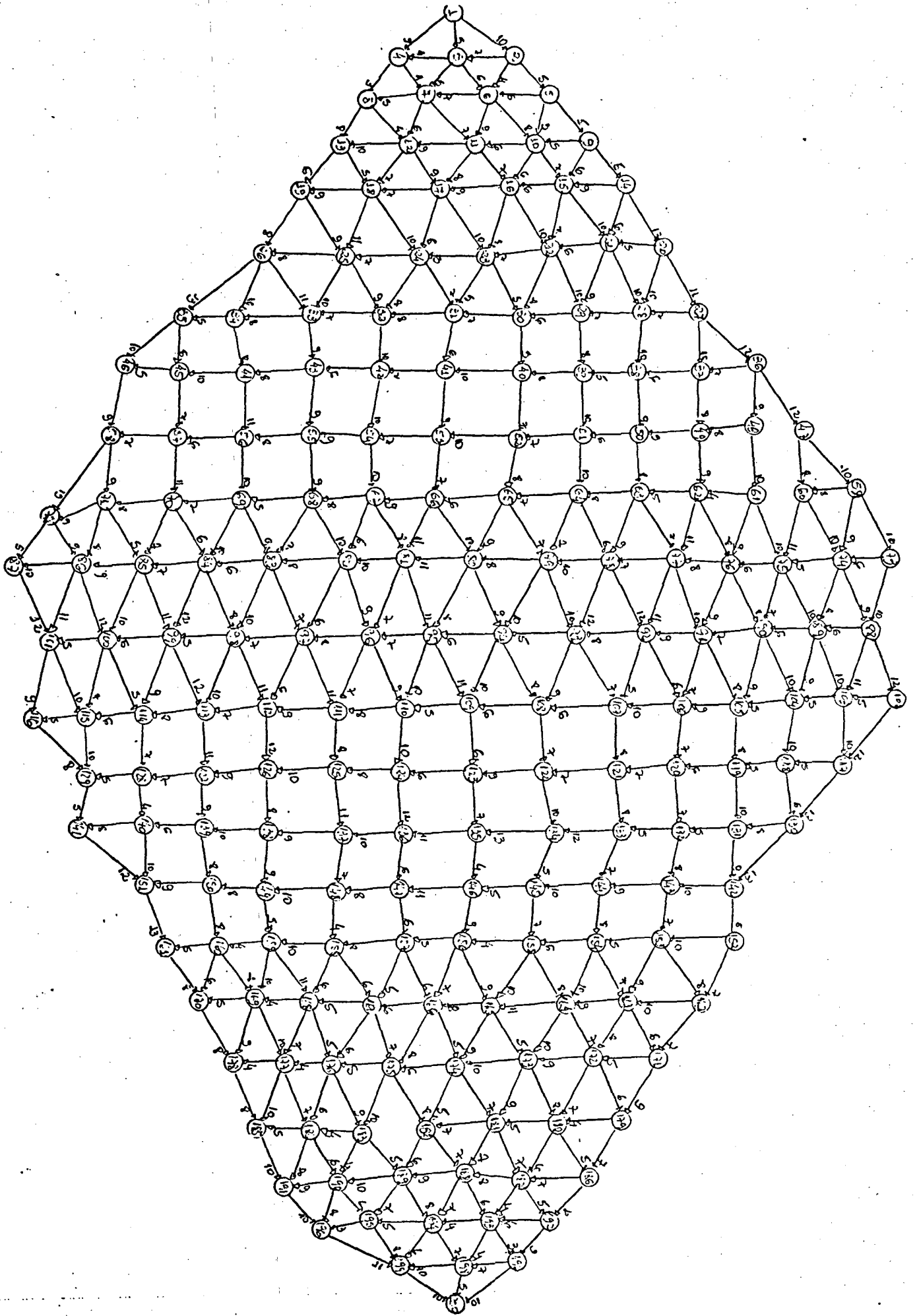


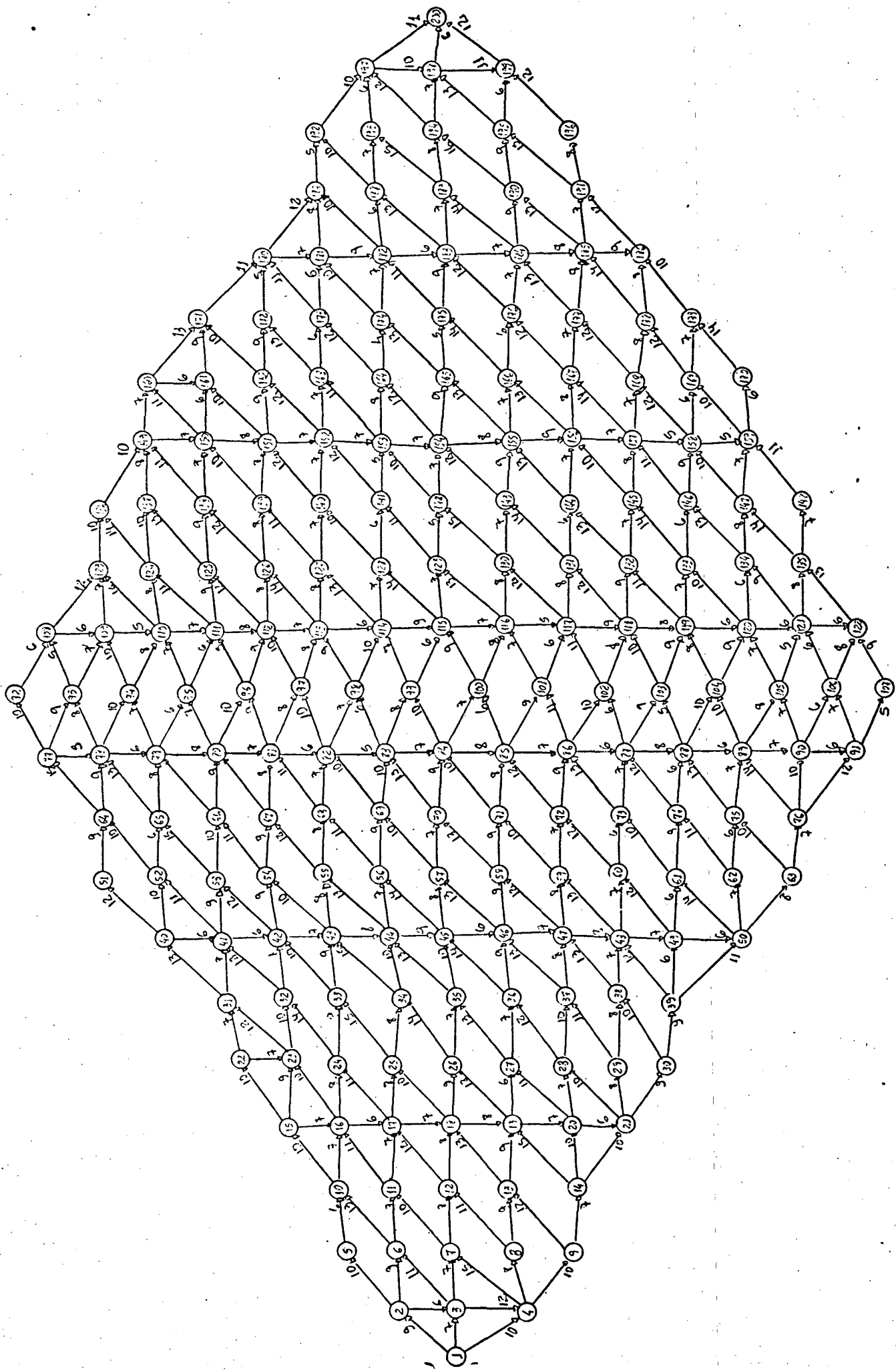
F I G U R A 16

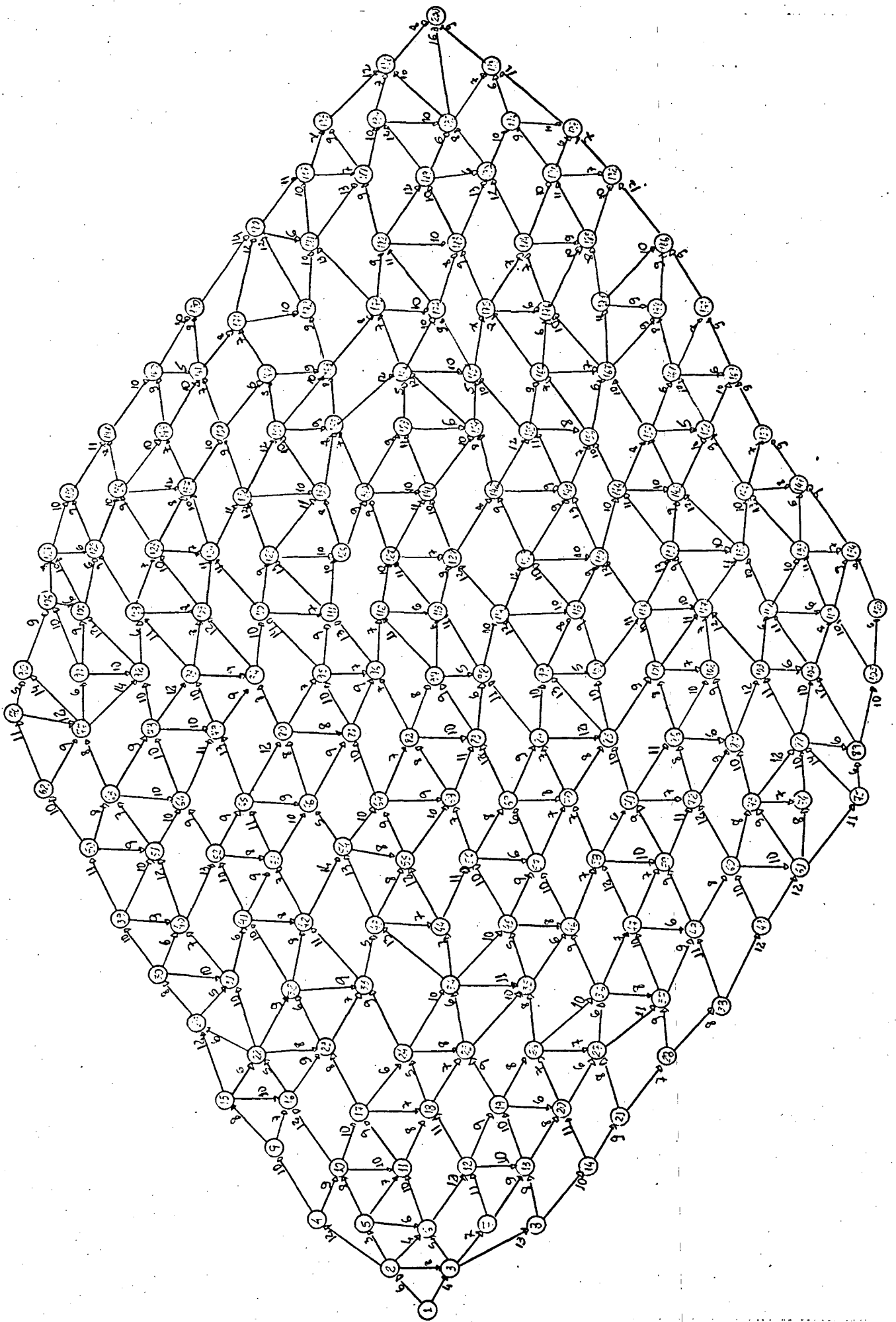
GRAFOS COM 200 VÉRTICES

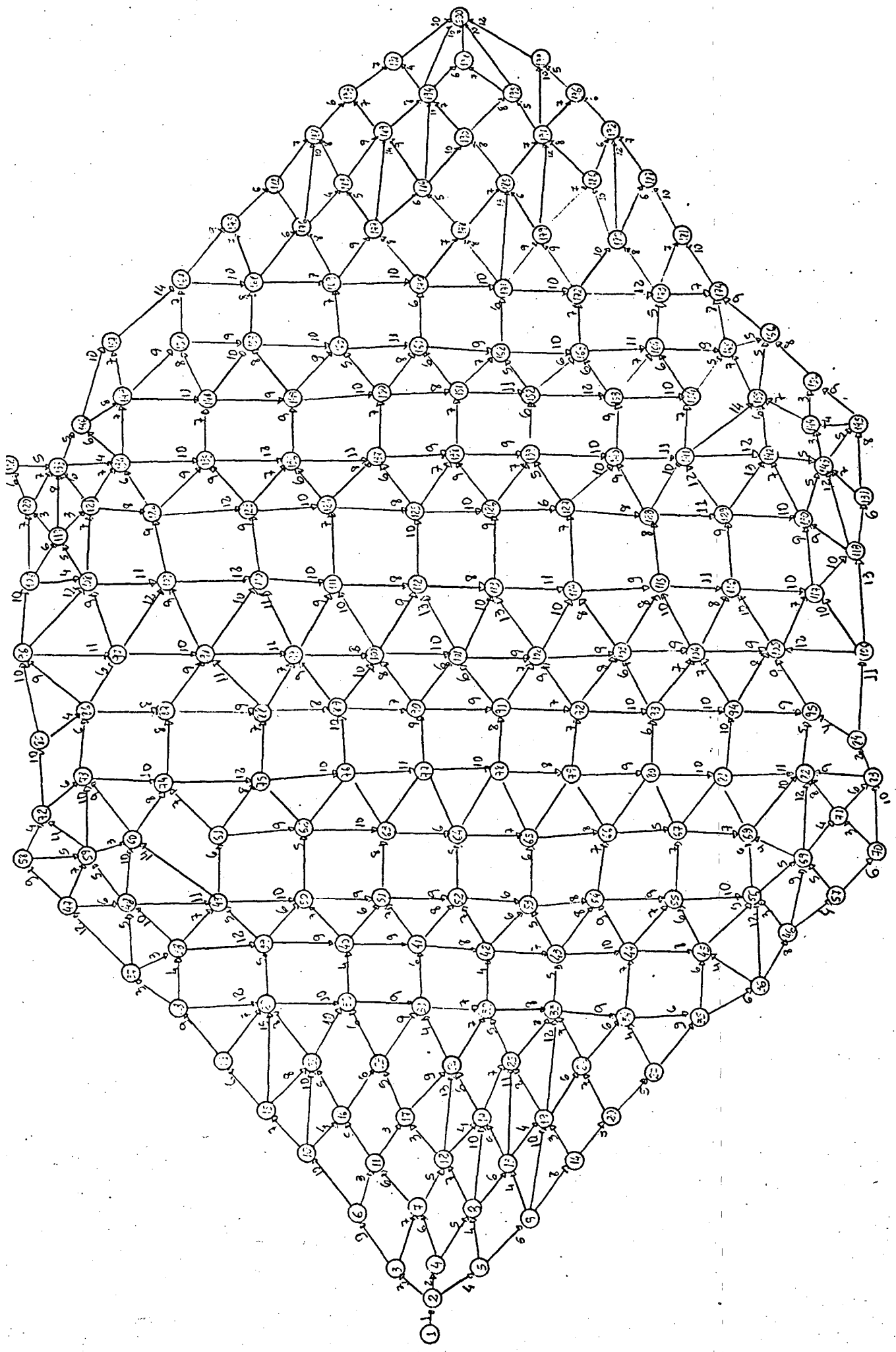


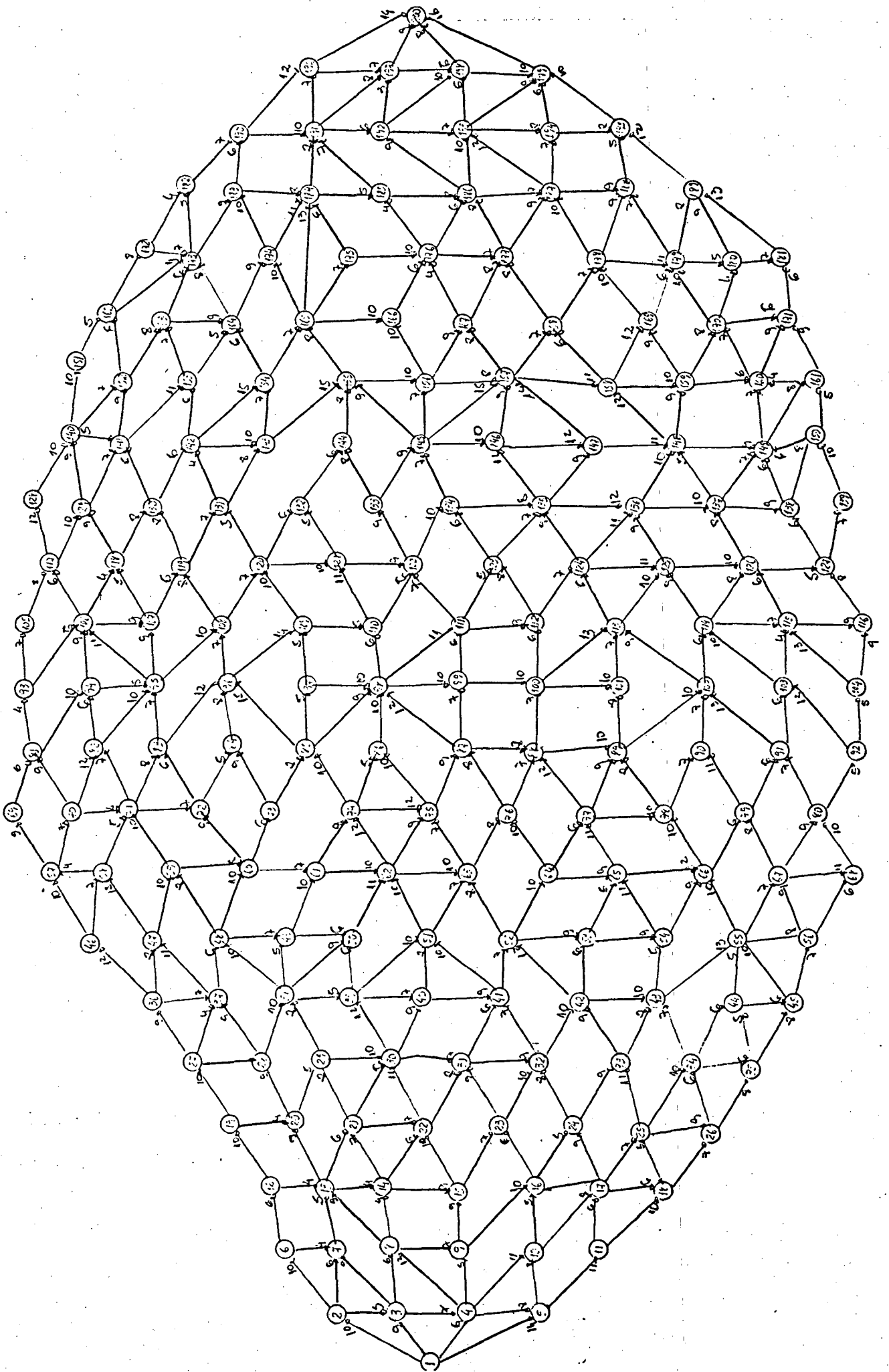


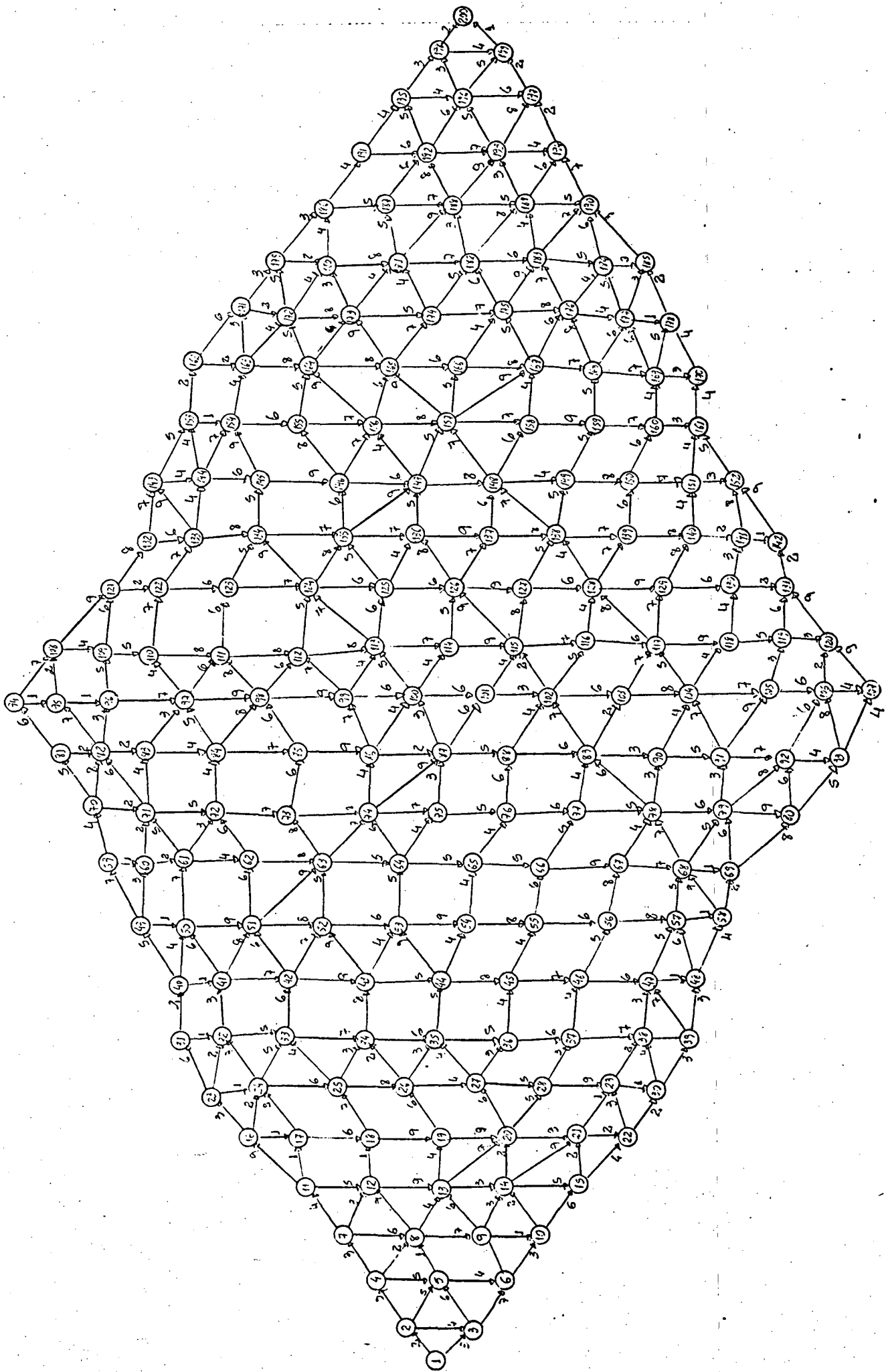


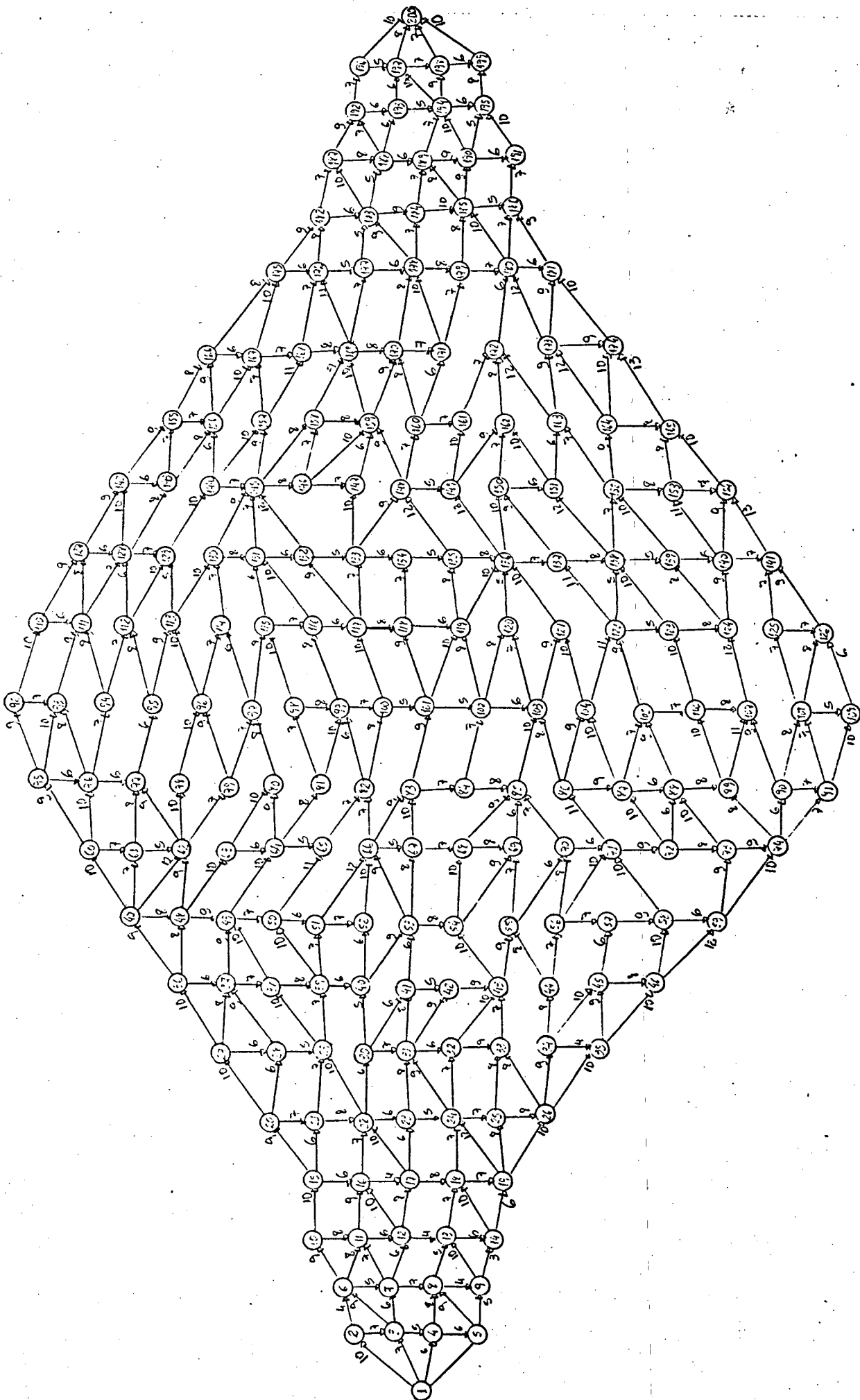


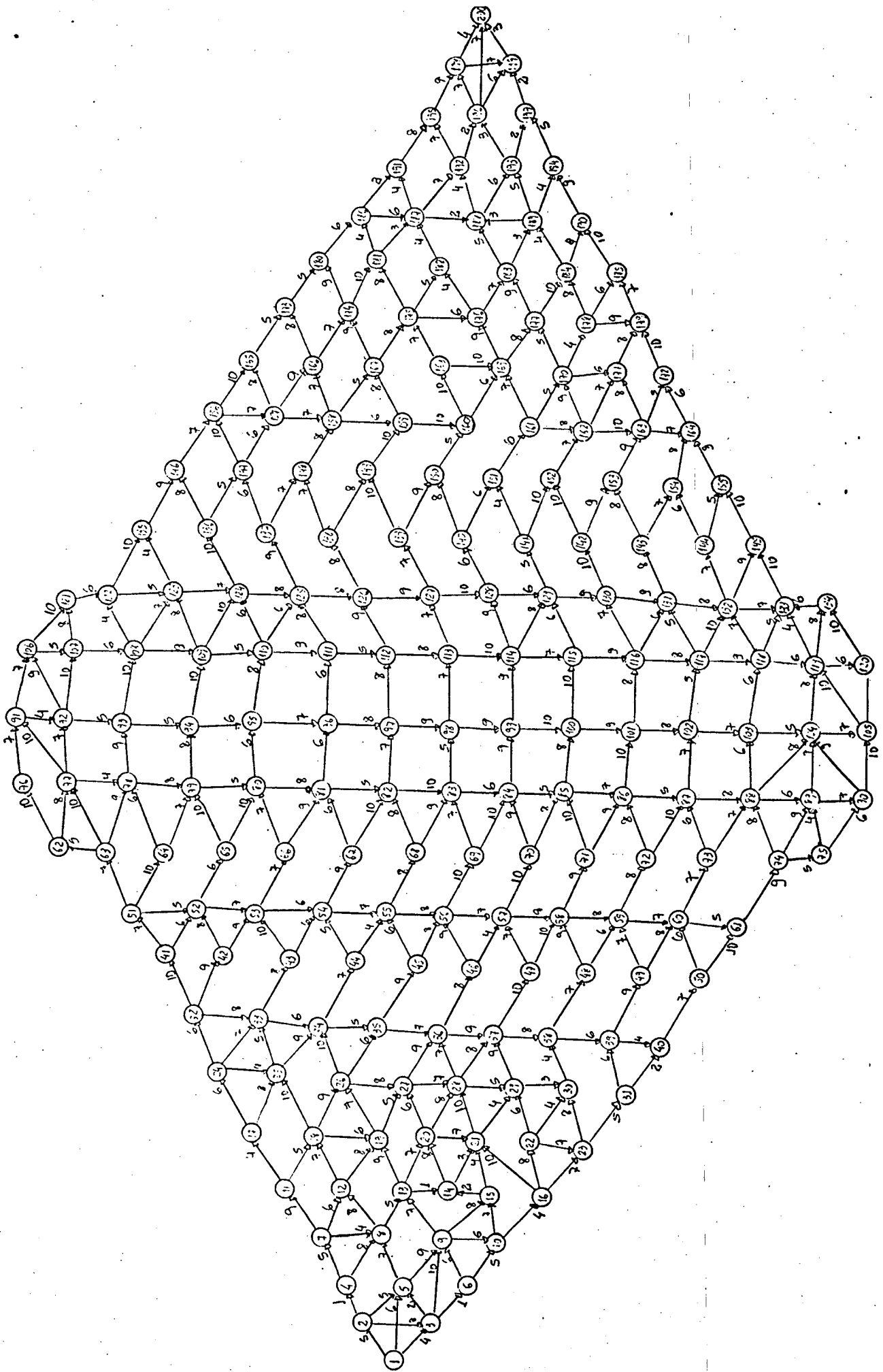


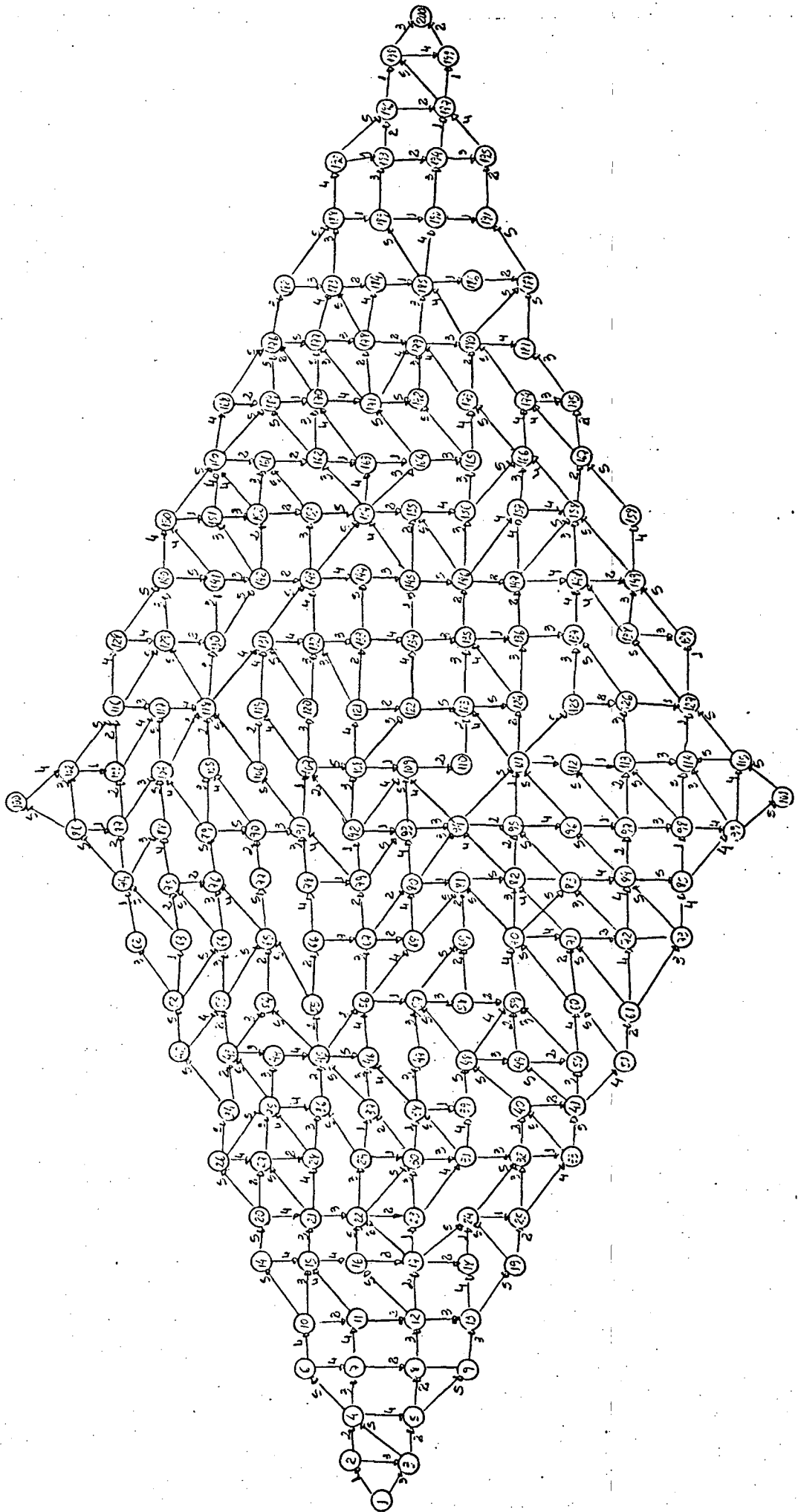


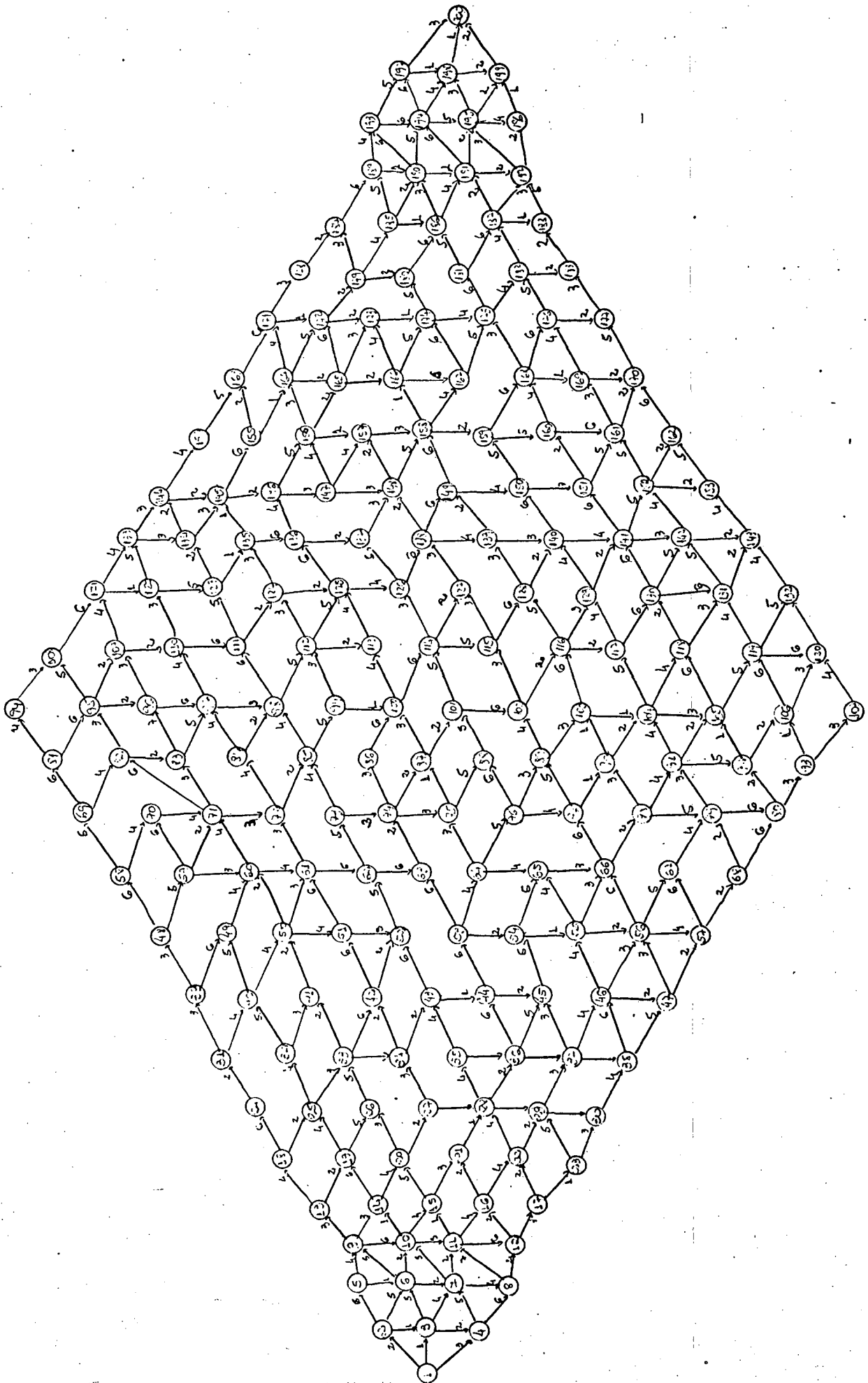


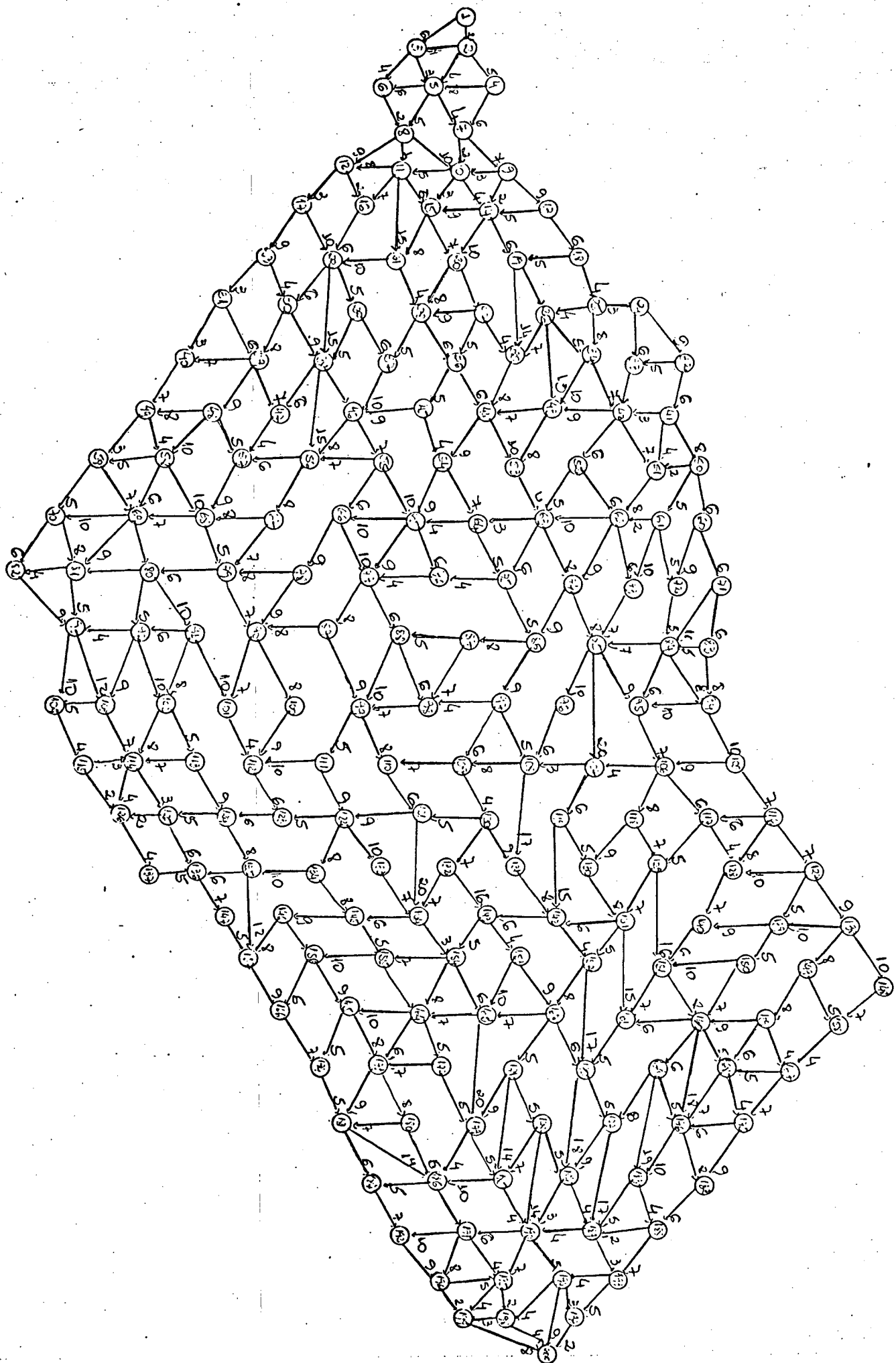


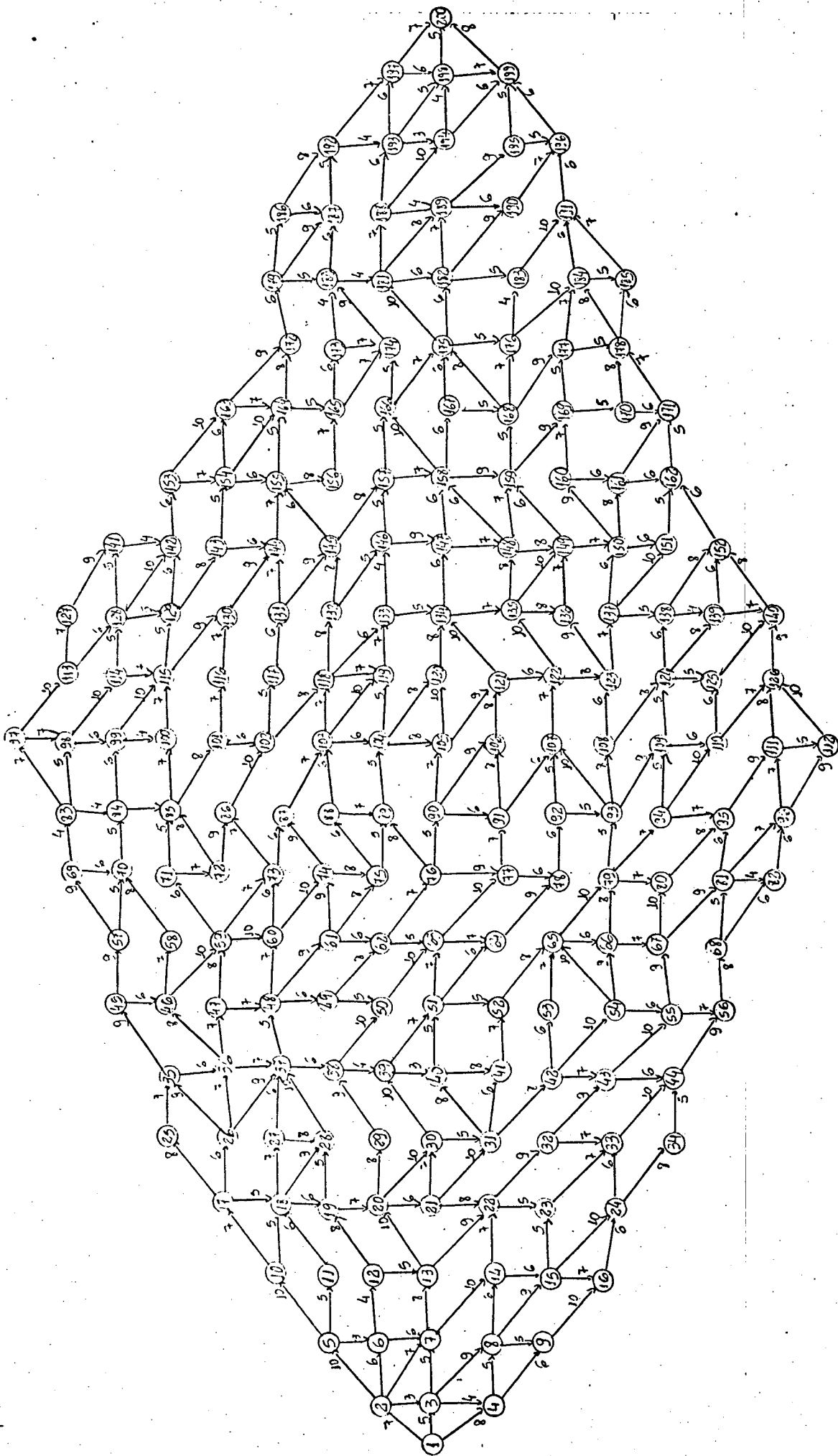


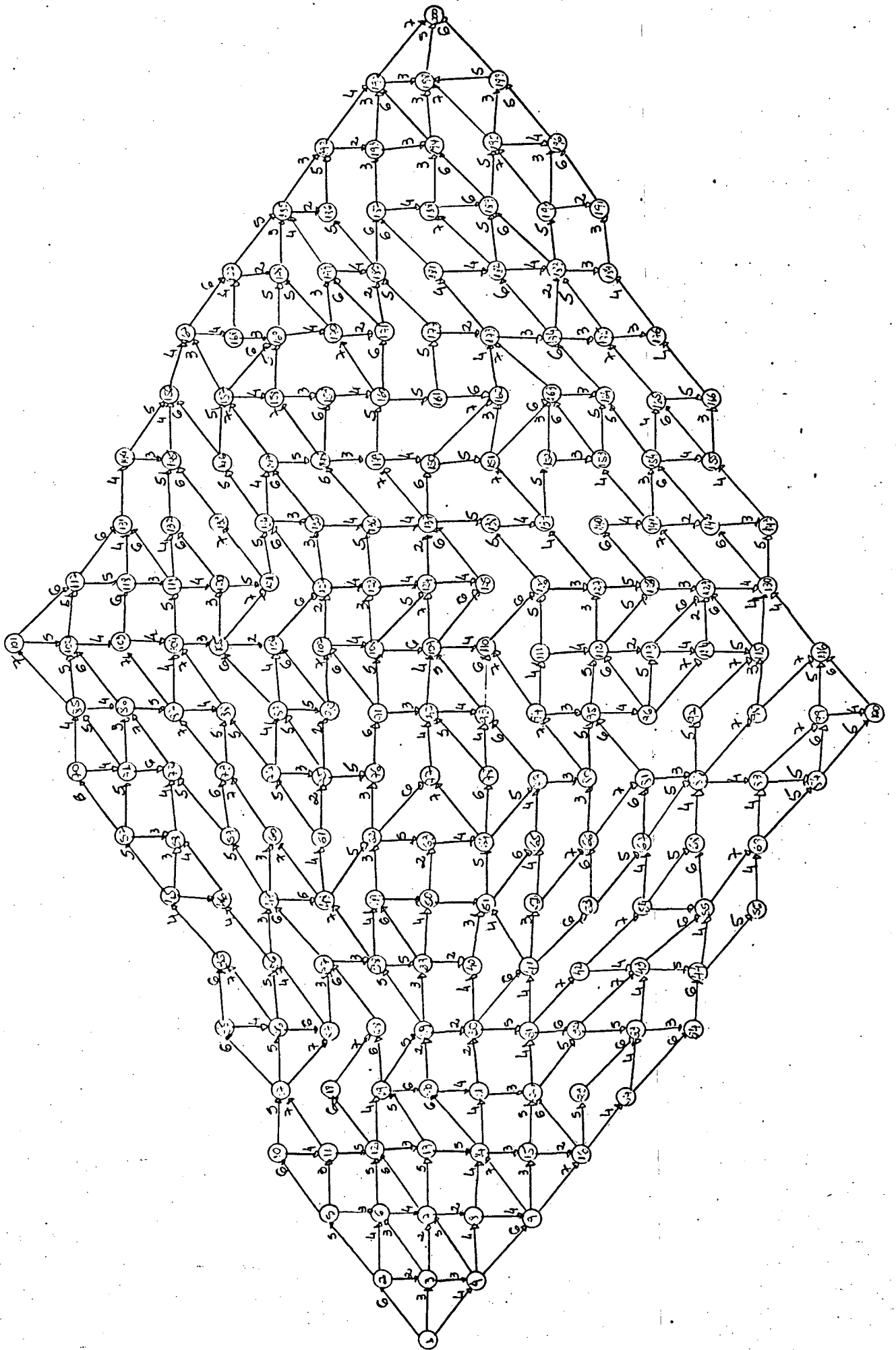


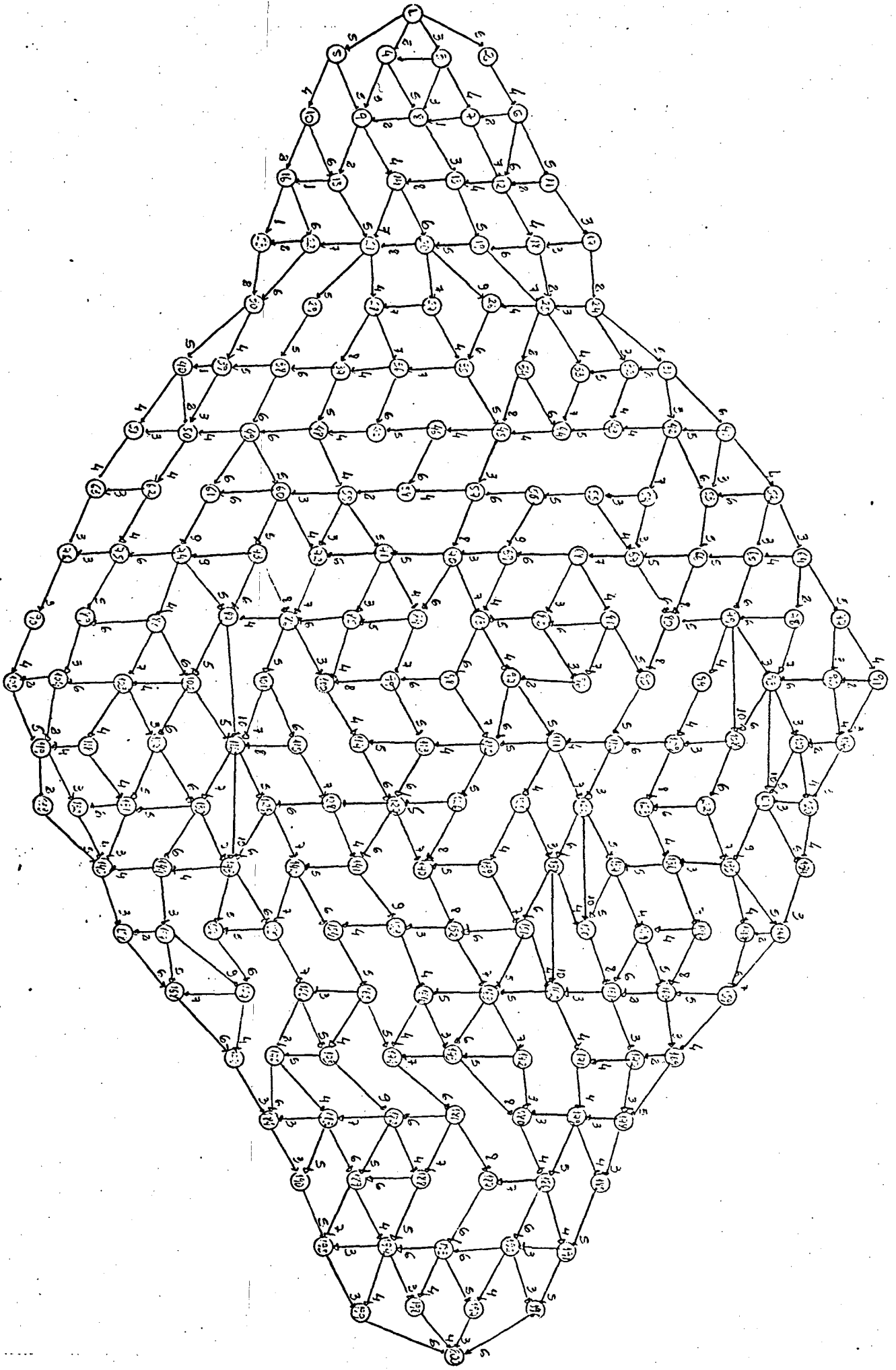


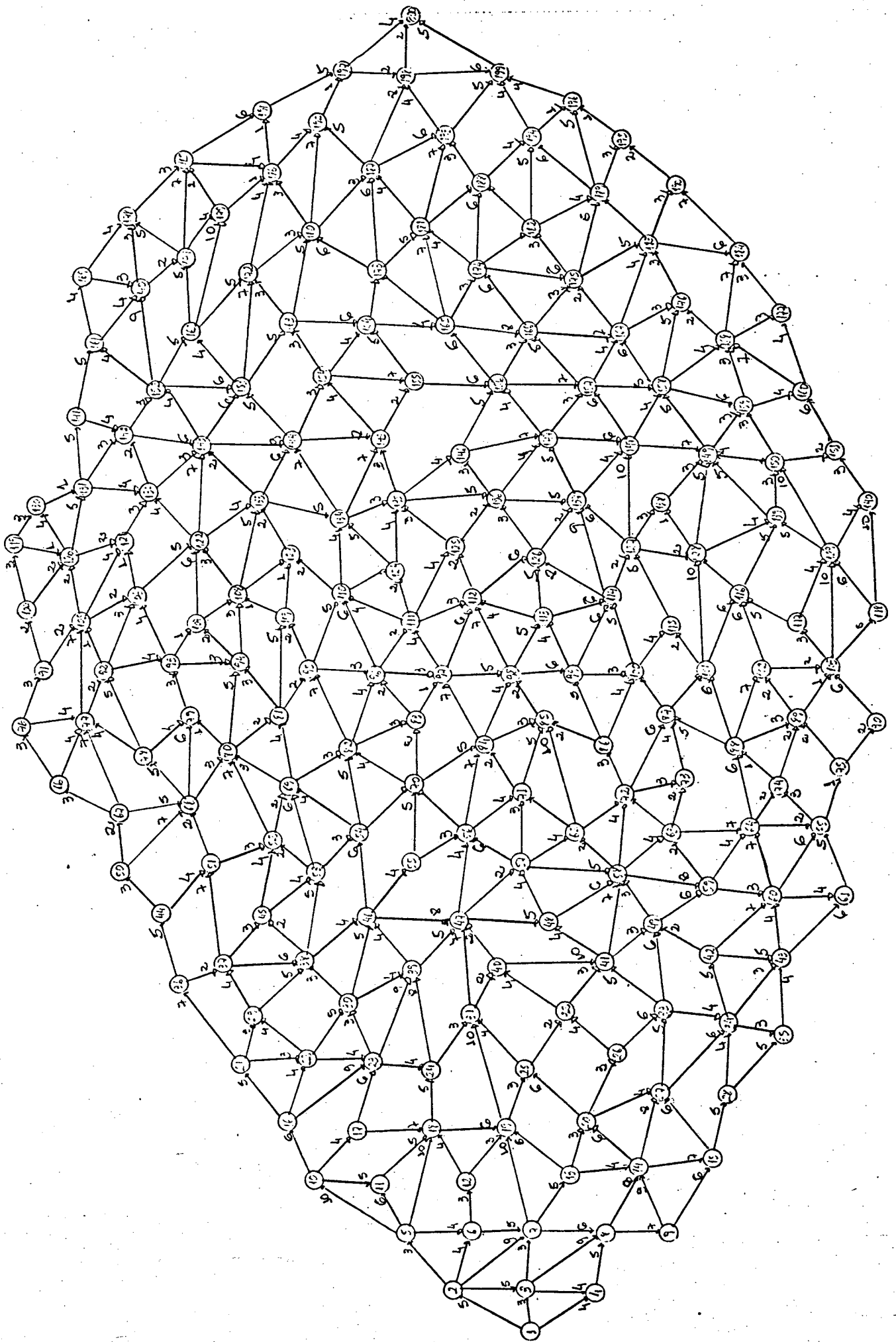


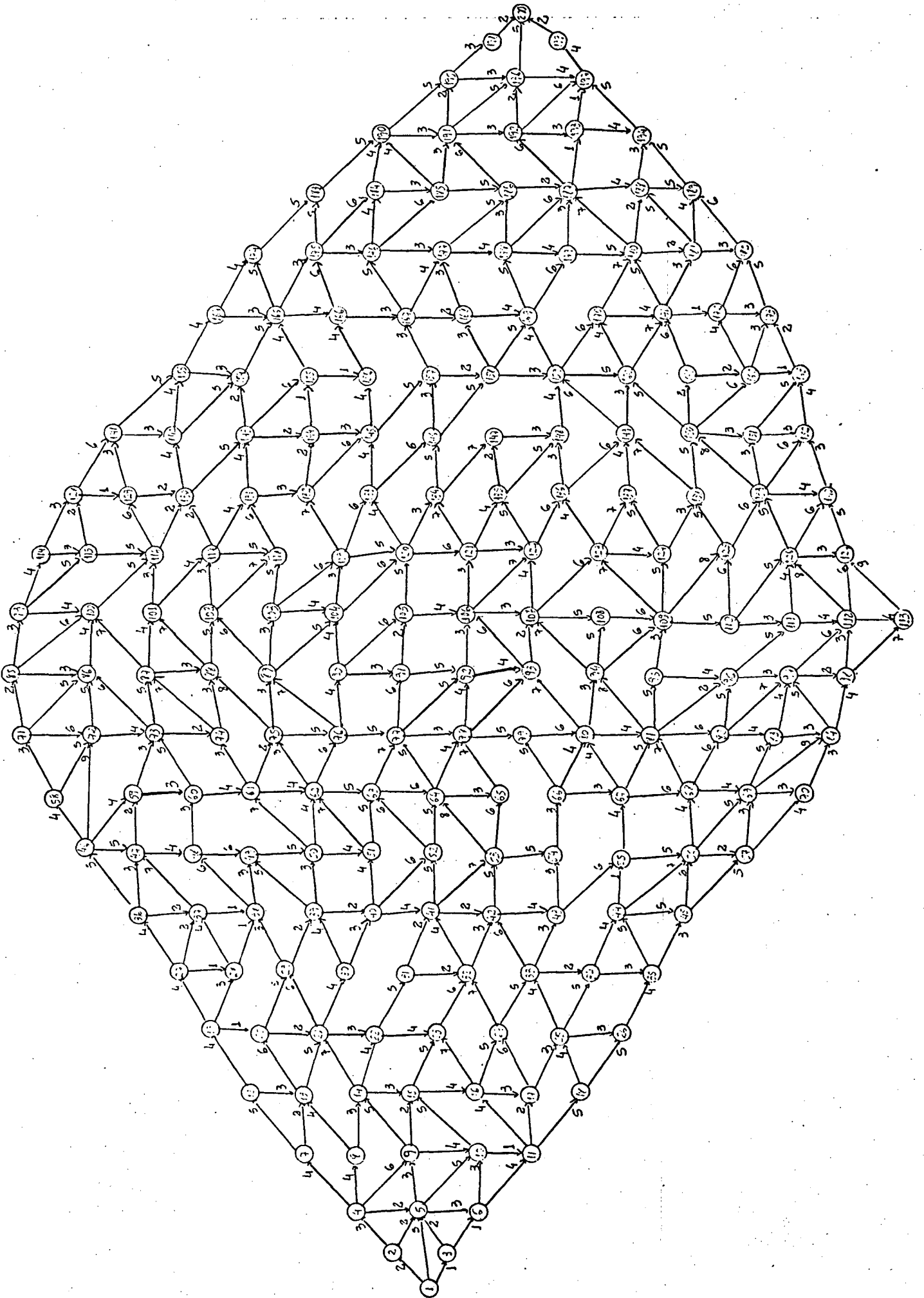


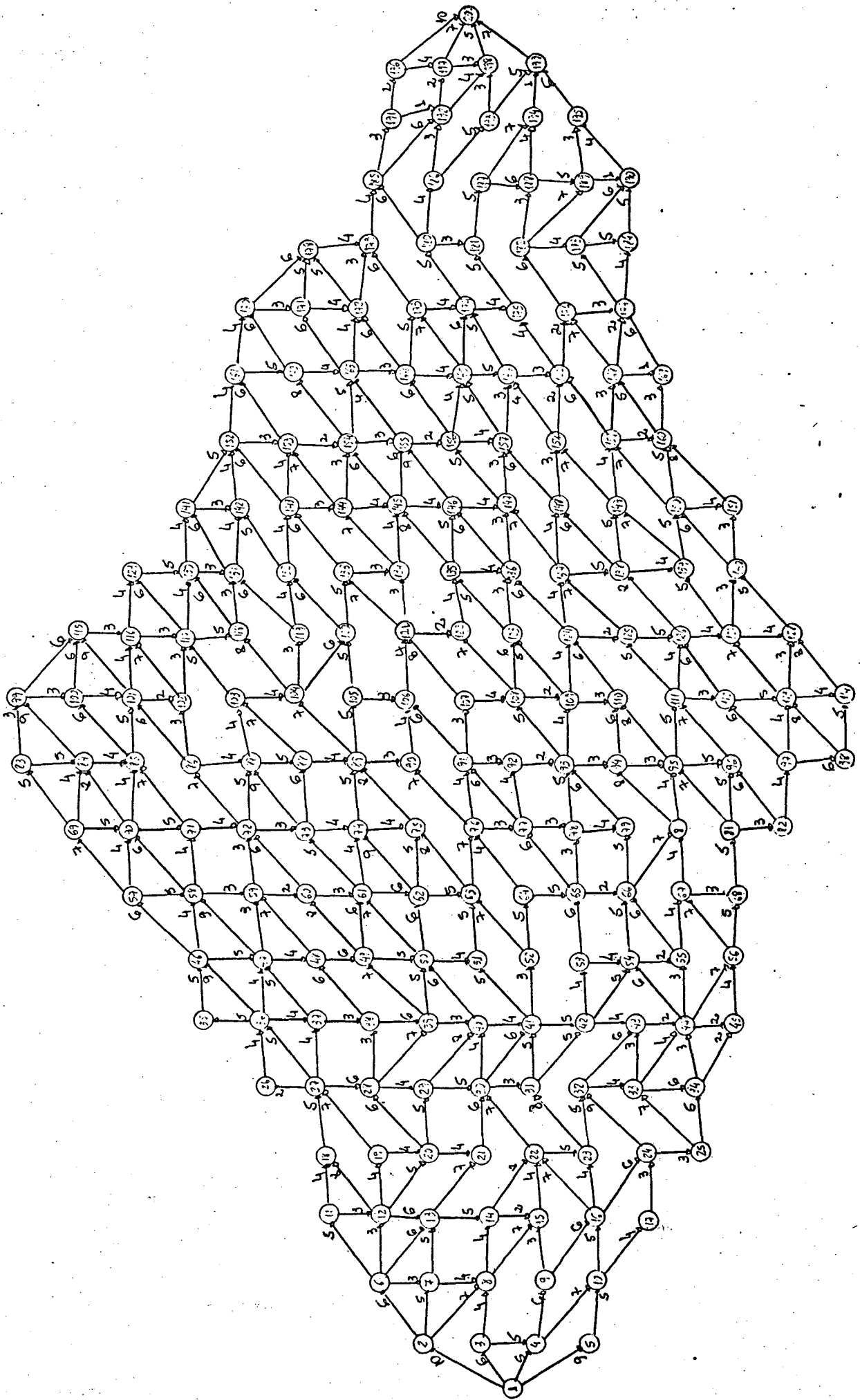












A N E X O 2

PROGRAMAS DOS ALGORITMOS

PROGRAMA DE

DIJKSTRA

```

SJ03          TESTE20,KP=29
INTEGER D(200,200),PAI(200),GCHAP(200),FEXADO(200),SOMA,FLINHA,FC  TES00010
*HAP,INDF(200),ORD,VIA(200),CAM(200),AUXG(200),CRDI  TES00020
C                                                    TES00030
C *****
C                                                    TES00040
C *****
C          ALGORITMO DE DIJKSTRA
C *****
C          PARA UMA SERIE DE 20 PROBLEMAS
C *****
C          TES00080
C          TES00090
C          =====
C          TES00100
C          ==          N - NUMERO DE VERTICES DO GRAFO          == TES00110
C          ==          NP - NUMERO DE ORDEM DO PROBLEMA          == TES00120
C          ==          NCC - NUMERO DE CARTOES COMPLETOS          == TES00130
C          ==          NCI - NUMERO DE CARTOES INCOMPLETOS          == TES00140
C          ==          NNCI - NUMERO DE VERTICES EM CARTOES INCOMPLETOS == TES00150
C          ==          NPROB - NUMERO DE PROBLEMAS          ==
C          =====
C          TES00160
C *****
C          LEITURA DOS DADOS
C *****
C          TES00170
C          TES00180
C          TES00190
C
C          READ(5,38)NPROB,N
38  FORMAT(2I5)
C          WRITE(5,102)
C          WRITE(5,103)N
C
C          DO 110 IK=1,NPROB
C
C          READ(5,1)NP,NCC,NCI,NNCI
C          1  FORMAT(4I5)
C          TES00240
C          TES00260
C          TES00270
C *****
C          INICIALIZACAO DA MATRIZ PRINCIPAL DO GRAFO
C *****
C          TES00280
C          TES00290
C
C          IF(N.LT.1) GOTO 1114
C          DO 3 I=1,N
C            DO 2 J=1,N
C              D(I,J)=9999
C          2   CONTINUE
C          3   CONTINUE
C          TES00300
C          TES00310
C          TES00320
C          TES00330
C          TES00340
C          TES00350
C          TES00360
C *****
C          LEITURA EM CARTOES COMPLETOS
C *****
C          TES00370
C *****
C          TES00380
C          IF(NCC.LT.1) GOTO 6
C          DO 5 K=1,NCC
C            READ(5,4)(I,J,D(I,J),L=1,6)
C          4   FORMAT(6(I3,I3,I6))
C          TES00390
C          TES00400
C          TES00410
C          TES00420
C          TES00430
C          TES00440
C          5   CONTINUE

```

```

C *****
C ***** LEITURA EM CARTOES INCOMPLETOS ***** TES00450
C *****
C
6 IF(NCI.LT.1) GOTO 9 TES00460
  READ(5,7)(I,J,D(I,J),L=1,NNCI) TES00470
7   FORMAT(6(I3,I3,16)) TES00480
9 CONTINUE TES00490
  TES00500
  TES00510
C *****
C ***** CALCULO DO CUSTO MAXIMO DO GRAFO ***** TES00520
C *****
C
SOMA=0 TES00530
DO 11 I=1,N TES00570
  DO 10 J=1,N TES00580
    IF(D(I,J).GE.9999) GOTO 10 TES00590
    SOMA=SOMA+D(I,J) TES00600
10 CONTINUE TES00610
11 CONTINUE TES00620
  TES00630
C
C =====
C == FEXADO - VETOR QUE ARMAZENA AS INFORMACOES DOS VERTICES ==
C == FEXADO = 0 - VERTICES NAO-GERADOS ==
C == FEXADO = 1 - VERTICES GERADOS OU ABERTOS ==
C == FEXADO = 2 - VERTICES PESQUISADOS OU FECHADOS ==
C ===== TES00640
C
C =====
C == K - VERTICE ROTULADO COMO PERMANENTE == TES00650
C == PAI - VETOR QUE ARMAZENA OS VERTICES PERMANENTES == TES00660
C == NA - NUMERO DE ELEMENTOS QUANDO FEXADO = 1 == TES00670
C == NF - NUMERO DE ELEMENTOS QUANDO FEXADO = 2 == TES00710
C == INC - ATUAL INDICE DO VERTICE DE MENOR CUSTO == TES00720
C == G(CHAP) CUSTO MINIMO DO REFERIDO VERTICE == TES00730
C == NG - VERTICE ATUAL DE G(CHAP) == TES00740
C ===== TES00750
C ===== TES00760
C *****
C ***** CONDICoes INICIAIS *****
C *****
C
K = 1
IND = 1
PAI(K) = 0
NA=0 TES00770
C
DO 12 I=1,N TES00780
  FEXADO(I)=0 TES00790
  PAI(I)=0 TES00800
  CAM(I)=0 TES00810
  GCHAP(I)=0
12 CONTINUE TES00820
C
  FEXADO(K)=1 TES00830

```

```

        NA=NA+1
        NF=0
C
GCHAP(K)=0
GCHAP(N)=SOMA
NG=0
GO TO 13
C
*****
*****
C
*****
*****
C
*****
*****
C
135 IF(NFIL.NE.1) GO TO 13
        IND1=INDF(CRD)
        GO TO 133
13 IF(NA-1) 1111,20,14
14 IND1=AUXG(NG)
133 MINI=GCHAP(INC1)
        IND=IND1
        IF(FEXADD(INC).EQ.2) GO TO 201
        DO 18 I=2,N
            IF(I.EQ.IND) GO TO 17
            IF(FEXADD(I).NE.1) GO TO 16
            IF(GCHAP(I).GT.MINI) GO TO 15
            MINI=GCHAP(I)
            IND=AUXG(I)
15 CONTINUE
16 CONTINUE
17 CONTINUE
18 CONTINUE
C
20 FEXADD(IND)=2
        NA=NA-1
        NF=NF+1
C
        IF(NF.EQ.N) GO TO 1112
        GO TO 202
C
201 IND1=N
        GO TO 133
C
*****
*****
C
*****
*****
C
=====
C
==      INDP - INDICE DO VERTICE PERTENCENTE AO VETOR PAI      ==
C
==      INDF  - INDICE DO VERTICE SUCESSOR                    ==
C
==      NFIL - NUMERO DE SUCESSORES                            ==
C
==      ORD  - ORDEM DO VERTICE SUCESSOR                       ==
C
=====
C
202 IF(IND.EQ.N) GO TO 1112
        INDP=IND
        ORD=0

```

TES00840
 TES00850
 TES00880
 TES00900
 TES00910
 TES00920
 TES00930
 TES00940
 TES00950
 TES00960
 TES00970
 TES00980
 TES00990
 TES01000
 TES01010
 TES01020
 TES01030
 TES01040
 TES01050
 TES01060
 TES01070
 TES01080
 TES01090
 TES01100
 TES01110
 TES01120
 TES01130
 TFS01140
 TES01150
 TES01160
 TES01170
 TES01180
 TES01190
 TES01200
 TES01210
 TES01220
 TES01230
 TES01240
 TES01250
 TES01260
 TES01270
 TES01280
 TES01290
 TES01300
 TES01310
 TES01320
 TES01330
 TES01340
 TES01350

```

NFIL=0
DO 23 I=2,N
  IF(I.EQ.INDP) GO TO 22
  IF(D(INDP,I).GE.9999) GO TO 21
  ORD=ORD+1
  INDF(ORD)=I
  NFIL=NFIL+1
21 CONTINUE
22 CONTINUE
23 CONTINUE
*****
***** CALCULO DOS CUSTOS DOS SUCESSORES *****
*****
=====
== FLINHA - CUSTO AUXILIAR PARA CALCULO DE CADA GCHAP ==
== FCHAP - PROVAVEL CUSTO DE GCHAP ==
== AUXG - ARMAZENA OS 'IND' DE GCHAP ORDENADO ==
=====
IF(NFIL.EQ.0) GO TO 14
DO 25 I=1,NFIL
  FLINHA=GCHAP(IND)+D(IND,INDF(I))
  IF((FEXADO(INDF(I)).EQ.1).AND.(GCHAP(INDF(I)).LT.FLINHA))
*0 TO 24
  IF(FEXADO(INDF(I)).EQ.2) GO TO 24
  FCHAP=FLINHA
  GCHAP(INDF(I))=FCHAP
  AUXG(INDF(I))=INDF(I)
  PAI(INDF(I))=IND
  FEXADO(INDF(I))=1
  NA=NA+1
  NG=INDF(I)
24 CONTINUE
25 CONTINUE
GO TO 135
*****
***** RECONSTITUICAO DO CAMINHO DE MINIMO CUSTO DO GRAFO *****
*****
=====
== IC -- NUMERO DE VERTICES DO CAMINHO DE MINIMO CUSTO ==
== INDC -- INCICE QUE INDICAM O CAMINHO DE MINIMO CUSTO ==
== MIN -- CUSTO MINIMO FINAL DO GRAFO ==
== CAM -- VETOR QUE ARMAZENA O CAMINHO FINAL DO GRAFO ==
== VIA -- VETOR QUE ARMAZENA O CAMINHO FINAL ORDENADO ==
=====
1112 IC=1
INDC=N
MIN=0
DO 32 I=1,NF
DO 31 J=1,N

```

TES01360
 TES01370
 TES01380
 TES01390
 TES01400
 TES01410
 TES01420
 TES01430
 TES01440
 TES01450
 TES01460
 TES01470
 TES01480
 TES01490
 TES01500
 TES01520
 TES01540
 TES01550
 TES01560
 TES01570
 TES01580
 TES01590
 TES01600
 TES01620
 TES01630
 TES01640
 TES01650
 TES01660
 TES01670
 TES01680
 TES01690
 TES01700
 TES01710
 TES01730
 TES01740
 TES01750
 TES01760
 TES01770
 TES01790
 TES01790
 TES01800
 TES01810
 TES01820
 TES01830
 TES01840
 TES01850
 TES01860


```

                IF(PAI(INDC).NE.J) GO TO 30
                CAM(IC)=INDC
                MIN=MIN+D(J,INDC)
                IC=IC+1
                INDC=J
30              CONTINUE
31              CONTINUE
32              CONTINUE
C
                DO 33 I=1, IC
                JJ=IC+1-I
                VIA(JJ)=CAM(I)
33              CONTINUE
                VIA(1)=1
                GO TO 1113
C
C *****
C *****
C *****
C *****
1111 WRITE(5,100)
100 FORMAT(1H1,1X,37(' '),/,4X,'ALGORITMO DE DIJKSTRA',/,2X,37(' '))
    WRITE(5,101)
101 FORMAT(///,5X,'BUSCA EM GRAFO FRACASSADA',//)
    GO TO 1114
102 FORMAT(1H1,1X,37(' '),/,2X,'***** ALGORITMO DE DIJKSTRA *****',
/,2X,37(' '))
103 FORMAT(2X,'NUMERO DE VERTICES DO GRAFO = ',15,/,2X,30(' '))
1113 WRITE(5,104)NP
104 FORMAT(/,2X,'*** PROBLEMA NO. = ',15,' ***')
    WRITE(5,105)MIN
105 FORMAT(/,2X,'CUSTO MINIMO DO GRAFO = ',15)
    WRITE(5,106)
106 FORMAT(/,2X,'CAMINHO DE MINIMO CUSTO DO GRAFO =')
    WRITE(5,107)(VIA(J),J=1,IC)
107 FORMAT((2X,15(15)))
    WRITE(5,108)
108 FORMAT(2X,80(' '))
110 CONTINUE
1114 STOP
    END
$ENTRY

```

TES01870
 TES01880
 TES01890
 TES01900
 TES01910
 TES01920
 TES01930
 TES01940
 TES01950
 TES01960
 TES01970
 TES01980
 TES01990
 TES02000
 TES02010
 TES02020
 TES02030
 TES02040
 TES02050
 TES02060
 TES02070
 TES02080
 TES02090
 TES02110
 TES02150
 TES02120
 TES02130
 TES02170
 TES02180
 TES02200
 TES02210
 TES02220
 TES02230
 TES02270
 TES02280

P R O G R A M A D E

J E Y A R A T N A M

FILED ALGO WATFIV A1 VM/SP R4 PUT8602+ SLU408 CMS - LFSC

```

JOB          ALGO45,KP=29
INTEGER D(200,200),PI(200),FI(200),PIBARR(200),FIBARR(200),FEXADO
*(200),SUMA,CONT,VIA(200),PAI(200),FILHO(200),ALX(200),CAM(200),SO
*S(200),ORD,ORCI
C
C *****
C *****
C          ALGORITMO DE 'S. JEYARATNAM'
C          PARA UMA SERIE DE 20 PROBLEMAS
C *****
C *****
C *****
C          N - NUMERO DE NOS DO GRAFO
C          NCC - NUMERO DE CARTOES COMPLETOS
C          NCI - NUMERO DE CARTOES INCOMPLETOS
C          NNCI - NUMERO DE NOS DO CARTAO INCOMPLETO
C          NPRUB - NUMERO DE PROBLEMAS
C *****
C *****
C          LEITURA DOS DADOS
C *****
C *****
C          READ(5,1)NPROB,N
1  FORMAT(2I5)
   WRITE(5,100)
   WRITE(6,101)N
C
C          DU 110 IK=1,NPROB
C
C          READ(5,2)NP,NCC,NCI,NNCI
2  FOPMAT(4I5)
C *****
C *****
C          INICIALIZACAO DA MATRIZ PRINCIPAL DO GRAFO
C *****
C *****
C          IF(N.LT.1) GOTO 1001
   DU 4 I=1,N
     DO 3 J=1,N
       D(I,J)=9999
3  CONTINUE
4  CONTINUE
C *****
C *****
C          LEITURA EM CARTOES COMPLETOS
C *****
C *****
C          IF(NCC.LT.1) GOTO 7
   DO 5 K=1,NCC
     READ(5,5)(I,J,D(I,J),L=1,6)
5     FURMAT(6(I3,I3,16))
6  CONTINUE

```

```

C
C
C *****
C ***** LEITURA EM CARTOES INCOMPLETOS *****
C *****
C
C
C 7 IF(NCI.LT.1) GOTO 9
C READ(5,8)(I,J,D(I,J),L=1,NNCI)
C 8 FORMAT(6(I3,I3,I6))
C 9 CONTINUE
C
C *****
C ***** CALCULO DO CUSTO MAXIMO DO GRAFU *****
C *****
C
C SOMA=0
C DO 11 I=1,N
C DO 10 J=1,N
C IF(D(I,J).GE.9999) GOTO 10
C SOMA=SOMA+D(I,J)
C 10 CONTINUE
C 11 CONTINUE
C
C
C =====
C == FEXADO - VETOR QUE ARMAZENA AS INFORMACOES DOS VERTICES ==
C == FEXADO = 0 - VERTICES NAO-GERADOS ==
C == FEXADO = 1 - VERTICES GERADOS OU ABERTOS ==
C == FEXADO = 2 - VERTICES PESQUISADOS OU FECHADOS ==
C =====
C
C
C =====
C == DEFINICAO DOS CAMINHO DO GRAFO ( INICIO - FIM ) ==
C == FI - CUSTO DOS NOH'S (INICIO - FIM) ==
C == FIBARR - CUSTO DOS NOH'S JA PESQUISADOS ==
C =====
C
C DO 12 I=2,N
C PI(I)=D(1,I)
C 12 CONTINJE
C
C
C =====
C == DEFINICAO DOS CAMINHO DO GRAFO ( FIM - INICIO ) ==
C == FI - CUSTO DOS NOH'S (FIM - INICIO) ==
C == FIBARR - CUSTO DOS NOH'S JA PESQUISADOS ==
C =====
C
C NMI=N-1
C DO 13 I=1,NMI
C FI(I)=D(I,N)
C 13 CONTINJE
C
C DO 14 I=1,N
C FEXADD(I)=0
C 14 CONTINJE
C

```

ALG00380
ALG00390
ALG00400
ALG00410
ALG00420
ALG00430
ALG00440
ALG00450
ALG00460
ALG00470
ALG00500
ALG00510
ALG00520
ALG00530
ALG00540
ALG00550
ALG00560
ALG00570
ALG00580
ALG00590
ALG00600
ALG00610
ALG00620
ALG00630
ALG00640
ALG00650
ALG00660
ALG00670
ALG00680
ALG00690
ALG00700
ALG00710
ALG00720
ALG00730
ALG00770
ALG00780
ALG00790
ALG00800

```

C =====
C == PAI ARMAZENA O PROVAVEL CAM. DE MIN. CUSTO (INICIO-FIM) == ALG00810
C == FILHO ARMAZENA O PROVAVEL CAM. DE MIN. CUSTO (FIM-INICIO) == ALG00820
C == CONT - NUMERO DE VERTICES ROTULADOS COMO PERMANENTES == ALG00830
C == AUX - ARMAZENA OS INDICES DO VETOR PAI ==
C =====
C
C DO 15 I=1,N ALG00840
C PAI(I)=0 ALG00850
C FILHO(I)=0 ALG00860
C CAM(I)=0 ALG00870
15 CONTINUE ALG00880
C ALG00890
C *****
C ***** CONDIÇÕES INICIAIS ***** ALG00900
C ***** ALG00910
C ***** ALG00940
C K=1 ALG00950
C L=N ALG00960
C M=1 ALG00970
C CONT=2 ALG00990
C MIN = SOMA ALG00990
C PIBARR(K)=0 ALG01000
C FIBARR(L)=0 ALG01010
C FEXADD(K)=2 ALG01020
C FEXADO(L)=2 ALG01030
C PI(N)=SOMA ALG01040
C FI(1)=SOMA ALG01050
C NPI=0 ALG01060
C NFI=0 ALG01070
C ALG01080
C *****
C ***** DETERMINAÇÃO DO CAMINHO ENTRE VERTICES ***** ALG01090
C ***** NÃO-CONSECUTIVOS (INICIO-FIM) ***** ALG01100
C *****
C ***** ALG01120
16 DO 20 I=2,N ALG01150
C IF(FEXADO(I).EQ.2) GO TO 17 ALG01160
C IF(D(K,I).GE.9999) GO TO 18 ALG01170
C IF(PI(I).LT.PIBARR(K)+D(K,I)) GOTO 17 ALG01180
C PI(I)=PIBARR(K)+D(K,I) ALG01190
C PAI(I)=K ALG01200
C FEXADO(I)=1
C AUX(I)=I ALG01210
C ORD=AUX(I) ALG01220
17 CONTINUE ALG01230
18 CONTINUE ALG01240
19 CONTINUE ALG01250
20 CONTINUE ALG01260
C ALG01270
C *****
C ***** DETERMINAÇÃO DO PROXIMO VERTICE A SER ***** ALG01280
C ***** ROTULADO COMO PERMANENTE (INICIO-FIM) *****
C ***** ALG01290

```

```

31 IF(ORD.LT.N) GO TO 311
    IF(FEXADO(CRD).EQ.2) GO TO 1000
311 MINIMO=PI(ORD)
    KK=ORD
    IF(KK.EQ.N) GO TO 1000
    DO 24 I=2,N
        IF(I.EQ.KK) GO TO 23
        IF(FEXACO(I).NE.1) GO TO 221
        IF(FEXACO(I).EQ.2) GOTO 22
            IF(PI(I).GE.MINIMO) GOTO 21
                MINIMO=PI(I)
                KK=AUX(I)
    21 CONTINUE
    22 CONTINUE
    221 CONTINUE
    23 CONTINUE
    24 CONTINUE
C
    IF(FEXADO(KK).NE.2) GO TO 30
        ORD=ORD+1
        GO TO 311
C
    30 IF(MIN.LE.PI(KK)) GOTO 1000
        K=KK
        PIBARR(K)=PI(KK)
        FEXADO(K)=2
        CONT=CONT+1
        NPI=NPI+1
C
C *****
C ***** DETERMINACAO DO CAMINHO ENTRE K E L (INICIO-FIM) *****
C *****
C
C IF(FI(K).LE.FIBARR(L)+D(K,L)) GOTO 25
    FI(K)=FIBARR(L)+D(K,L)
    FILHO(K)=L
    25 CONTINUE
C
C *****
C ***** DETERMINACAO DO PONTO DE ENCONTRO ENTRE OS *****
C ***** VETORES PAI E FILHO (INICIO-FIM) *****
C *****
C
    IF(MIN.LE.PIBARR(K)+FI(K))GOTO 26
    MIN=PIBARR(K)+FI(K)
    M=K
    26 CONTINUE
C
    IF(CONT.EQ.N) GOTO 1000
    GO TO 27
C
C *****
C ***** DETERMINACAO DO CAMINHO ENTRE OS VERTICES *****
C ***** NAO CONSECUTIVOS (FIM-INICIO) *****
C *****

```

ALG01300
ALG01310
ALG01320
ALG01330
ALG01340
ALG01350
ALG01360

ALG01370
ALG01380
ALG01390
ALG01400
ALG01410
ALG01420

ALG01430
ALG01440
ALG01450
ALG01480
ALG01490
ALG01500
ALG01510
ALG01520
ALG01530
ALG01540
ALG01550
ALG01560
ALG01570
ALG01580
ALG01590
ALG01600

ALG01610
ALG01620
ALG01630
ALG01640
ALG01650
ALG01660

ALG01670
ALG01680

ALG01690
ALG01700
ALG01710
ALG01720
ALG01730
ALG01740
ALG01750
ALG01760

ALG01770
ALG01780

```

C
27 DO 43 J=1,NM1
    IF(FEXADO(J).EQ.2) GO TO 42
    IF(D(J,L).GE.9999) GO TO 41
    IF(FI(J).LT.(FIBARR(L)+D(J,L))) GO TO 4C
    FI(J)=FIBARR(L)+D(J,L)
    FILHO(J)=L
    FEXADO(J)=1
    SGS(J)=J
    ORD1=SOS(J)
40     CONT INJE
41     CONTINUE
42     CONTINUE
43 CONTINUE
C
*****
*****      DETERMINACAO DO PROXIMO VERTICE A SER      *****
*****      ROTULADO COMO PERMANENTE (FIM-INICIO)      *****
*****
C
51 IF(ORD1.GE.1) GO TO 511
    IF(FEXADO(ORD1).EQ.2) GO TO 1000
511 MICRO=FI(ORD1)
    LL=ORD1
    IF(LL.EQ.1) GO TO 1000
    DO 47 J=1,NM1
        IF(J.EQ.L) GO TO 46
        IF(FEXADO(J).NE.1) GO TO 455
        IF(FEXADO(J).EQ.2) GOTO 45
        IF(FI(J).GE.MICRO) GOTO 44
        MICRO=FI(J)
        LL=J
44     CONT INJE
45     CONTINUE
455    CONTINUE
46     CONTINUE
47 CONTINUE
C
    IF(FEXADO(LL).NE.2) GO TO 50
    ORD1=ORD1-1
    GO TO 511
C
50 IF(MIN.LE.FI(LL)) GOTO 1000
    L=LL
    FIBARR(L)=FI(LL)
    FEXADO(L)=2
    CONT=CONT+1
    NFI=NFI+1
C
*****
*****      DETERMINACAO DO CAMINHO ENTRE L E K (FIM-INICIO) *****
*****
C
IF(PI(L).LE.PIBARR(K)+D(K,L)) GOTO 48
    PI(L)=PIBARR(K)+D(K,L)

```

ALG01790
ALG01800
ALG01810
ALG01820
ALG01830
ALG01840

ALG01850
ALG01860
ALG01870
ALG01880
ALG01890
ALG01900
ALG01910

ALG01920
ALG01930

ALG01940
ALG01950
ALG01960
ALG01970
ALG01980
ALG01990
ALG02000

ALG02010
ALG02020
ALG02030
ALG02040
ALG02050
ALG02060

ALG02070
ALG02080
ALG02090
ALG02120
ALG02130
ALG02140
ALG02150
ALG02160
ALG02170
ALG02180
ALG02190
ALG02200
ALG02210
ALG02220
ALG02230
ALG02240

ALG02250
ALG02260
ALG02270

IND=M	ALG02690
DO 59 J=1,NFI	ALG02700
DO 58 I=1,N	ALG02710
IF(FEXADO(I).NE.2) GO TO 66	ALG02730
IF(FILHO(IND).NE.1) GO TO 65	ALG02740
VIA(NC)=I	ALG02750
NC=NC+1	ALG02760
IND=I	ALG02770
55 CONTINUE	ALG02780
56 CONTINUE	ALG02790
58 CONTINUE	ALG02810
59 CONTINUE	ALG02820
C	ALG02830
NC=NC-1	ALG02840
C	ALG02860
C	*****
C	***** IMPRESSAO DOS RESULTADOS *****
C	*****
C	*****
100 FORMAT(IH1,1X,43(' '),/,2X,'***** ALGORITMO DE S. JEYAPATNAM	ALG02900
*****',/,2X,43(' '))	ALG02910
101 FORMAT(2X,'NUMERO DE VERTICES DO GRAFO = ',I5,/,2X,8C(' '))	ALG02950
WRITE(5,102)NP	ALG02920
102 FORMAT(/,2X,'*** PROBLEMA NO. =',I5,' ***')	ALG02930
WRITE(5,103)MIN	ALG02970
103 FORMAT(/,2X,'COSTO MINIMO DO GRAFO = ',I5)	ALG02980
WRITE(5,104)	ALG03000
104 FORMAT(/,2X,'CAMINHO DE MINIMO COSTO DO GRAFO C')	ALG03010
WRITE(5,105)(VIA(J),J=1,NC)	ALG03020
105 FORMAT((2X,15(I5)))	ALG03030
WRITE(5,106)	
106 FORMAT(2X,80(' '))	
110 CONTINUE	
1001 STOP	ALG03060
END	ALG03130
SENTRY	

A N E X O 3

RESULTADOS COMPUTACIONAIS DOS PROBLEMAS

P R O B L E M A S D E

D I J K S T R A

FILED TESTE10 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 10

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 9

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 4 5 7 10

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 13

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 6 8 10

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 13

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 4 7 8 9 10

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 3 7 10

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 4 6 9 10

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 10

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 3 5 6 8 10

*** PROBLEMA NO. = 7 ***

FILEO TESTE10 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 11

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 10

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 12

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 7 9 10

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 10

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 3 5 8 9 10

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 6 8 9 10

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 39

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 7 10

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 8 10

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 14

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 8 10

FILED TESTE10 SAIDA A1 VM/SP R4 PUT9602+ SLU408 CMS UFSC

*** PROBLEMA NO. = 14 ***
CUSTO MINIMO DO GRAFO = 16
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 4 5 8 9 10

*** PROBLEMA NO. = 15 ***
CUSTO MINIMO DO GRAFO = 12
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 5 6 9 10

*** PROBLEMA NO. = 16 ***
CUSTO MINIMO DO GRAFO = 14
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 4 6 8 9 10

*** PROBLEMA NO. = 17 ***
CUSTO MINIMO DO GRAFO = 15
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 7 9 10

*** PROBLEMA NO. = 18 ***
CUSTO MINIMO DO GRAFO = 9
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 4 6 8 10

*** PROBLEMA NO. = 19 ***
CUSTO MINIMO DO GRAFO = 14
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 5 9 10

*** PROBLEMA NO. = 20 ***
CUSTO MINIMO DO GRAFO = 22
CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 5 8 10

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 15

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 8 10 13 15

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 7 9 11 13 14 15

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 5 7 9 12 15

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 11 13 15

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 5 8 12 15

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 10 13 15

*** PROBLEMA NO. = 7 ***

FILE0 TEST015 SAIDA A1 VM/SP R4 PUT3602+ SLU408 CMS UFSC

CUSTO MINIMO DO GRAFO = 14

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 12 15

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 2 5 9 12 15

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 8 11 15

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 10 13 15

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 11 13 15

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 8 9 11 13 15

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 11 14 15

FILEO TESTE15 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - LFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 10 13 15

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 26

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 8 10 13 15

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 7 10 13 15

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 28

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 9 11 15

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 13

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 3 5 9 12 14 15

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 10 15

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 27

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 10 11 14 15

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 20

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 7 10 13 18 19 20

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 14 18 19 20

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 12 16 19 20

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 10 12 16 19 20

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 7 10 13 15 17 19 20

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 10 14 18 19 20

*** PROBLEMA NO. = 7 ***

FILED TESTE20 SAIDA A1 VM/SP R4 PUTS602+ SLU408 CMS UFSC

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 5 7 11 14 17 19 20

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 4 6 11 14 17 18 20

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 4 7 12 16 18 20

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 3 7 8 11 15 18 19 20

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 25

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 6 8 14 17 19 20

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 5 6 9 13 15 18 20

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 4 7 9 12 15 18 20

FILEO TESTE20 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 5 9 13 15 16 18 19 20

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 16

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 9 13 19 20

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 5 9 13 17 20

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 8 12 16 18 20

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 10 13 16 18 19 20

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 8 11 15 18 20

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 13 17 19 20

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 25

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 26

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 6 9 12 13 17 21 23 25

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 33

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 10 15 21 24 25

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 9 13 17 20 22 23 25

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 7 9 15 18 23 25

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 30

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 5 9 12 16 17 20 23 25

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 39

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 10 14 19 22 25

*** PROBLEMA NO. = 7 ***

FILEO TESTE25 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS UFSC

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 8 10 14 16 18 20 23 25

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 30

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 15 21 23 25

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 9 12 15 18 20 23 24 25

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 8 10 13 16 19 23 25

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 67

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 7 10 12 15 19 21 24 25

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 3 7 12 15 17 20 22 24 25

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 13 17 18 22 25

FILED TESTE25 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 61

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 10 14 19 23 25

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 69

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 7 11 16 20 23 25

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 12 15 17 21 23 24 25

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 43

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 8 14 21 24 25

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 61

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 10 17 19 22 25

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 33

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 5 6 10 14 19 22 24 25

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 30

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 12 14 18 21 23 25

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 30

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 27

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 12 16 20 24 27 28 29 30

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 95

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 14 17 20 23 24 28 30

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 5 7 10 12 16 20 24 27 30

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 45

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 14 19 24 27 29 30

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 16 21 23 27 29 30

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 6 9 12 18 20 22 24 27 29 30

*** PROBLEMA NO. = 7 ***

FILEO TESTE30 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 37

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 6 9 14 18 19 20 25 27 29 30

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 27

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 8 12 19 24 27 28 30

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 72

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 10 17 22 24 27 30

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 61

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 12 16 22 24 27 29 30

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 35

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 13 19 23 28 29 30

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 12 17 21 23 29 30

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 10 14 19 23 26 30

FILED TESTE3D SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 10 17 20 24 28 30

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 6 11 15 20 25 28 30

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 11 16 19 27 30

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 13 18 24 23 27 29 30

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 11 15 19 24 28 30

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 17 20 25 27 28 30

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 12 17 21 25 28 30

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 35

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 8 14 20 26 30 33 35

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 13 19 25 29 34 35

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 13 19 24 29 33 35

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 26

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 13 19 23 27 32 33 35

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 12 13 19 23 24 29 30 33 35

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 19 26 30 34 35

*** PROBLEMA NO. = 7 ***

CUSTO MINIMO DO GRAFO = 34

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 4 8 14 15 22 28 32 35

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 6 11 20 27 32 35

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 2 4 8 13 19 24 28 31 33 35

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 33

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 4 8 15 22 27 32 34 35

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 7 11 18 25 30 35

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 32

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 3 6 9 17 22 27 31 35

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 38

CAMINHO DE MINIMO CUSTO DO GRAFO =
1 4 3 6 10 15 22 28 32 35

FILEO TESTE35 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - LFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 34

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 3 5 8 13 14 22 25 31 33 34 35

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 48

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 16 20 25 29 33 35

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 6 9 15 21 27 31 33 35

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 106

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 7 12 19 25 31 34 35

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 41

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 9 15 21 26 30 33 35

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 37

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 15 22 28 32 35

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 32

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 20 25 29 33 35

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 40

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 14 20 25 29 34 38 40

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 53

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 8 11 15 20 25 29 32 38 40

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 44

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 9 13 19 25 30 34 38 40

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 8 13 22 29 34 37 39 40

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 9 13 19 26 31 35 38 40

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 5 11 16 25 31 35 37 38 40

*** PROBLEMA NO. = 7 ***

FILE0 TESTE40 SAIDA A1 VM/SP R4 PUT#602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 45

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 11 19 24 29 33 36 40

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 50

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 20 24 29 33 36 39 40

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 89

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 9 14 17 22 28 36 38 40

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 59

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 7 11 16 21 26 31 35 38 40

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 9 14 19 23 31 34 38 39 40

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 9 14 19 24 29 33 37 40

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 51

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 20 26 31 34 38 40

FILED TESTE40 SAIDA A1 VM/SP R4 PUT#602+ SLU408 CMS UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 16 22 28 32 36 39 40

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 50

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 19 23 28 32 35 37 39 40

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 16 22 27 32 36 38 40

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 5 9 13 18 23 28 35 38 40

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 59

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 5 11 15 21 27 33 39 40

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 53

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 10 14 19 26 31 35 38 40

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 6 9 12 17 22 28 33 36 38 40

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 45

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 58

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 14 20 32 36 40 43 45

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 8 12 16 20 24 28 33 37 41 43 44 45

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 68

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 5 13 24 30 35 39 42 44 45

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 73

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 7 10 15 27 33 38 42 45

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 13 18 23 29 34 39 43 45

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 64

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 7 11 15 20 30 34 38 41 45

*** PROBLEMA NO. = 7 ***

FILE0 TESTE45 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 60

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 7 12 16 22 29 34 39 42 44 45

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 17 22 30 35 40 43 45

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 44

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 9 13 17 23 28 34 39 42 44 45

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 66

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 12 17 23 31 36 40 45

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 13 19 24 31 36 40 43 45

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 60

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 7 11 15 20 25 31 37 41 44 45

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 10 15 20 26 33 38 41 44 45

FILEO TESTE45 SAIDA A1 VM/SP R4 PUTR602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 62

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 19 23 27 31 38 41 43 44 45

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 54

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 3 5 8 13 18 23 28 34 38 41 44 45

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 59

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 16 21 27 32 36 39 41 43 45

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 68

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 9 12 18 23 34 38 41 43 45

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 81

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 17 21 26 31 39 42 44 45

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 62

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 4 6 9 14 18 25 31 37 41 44 45

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 66

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 7 15 22 33 37 40 41 44 45

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 50

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 37

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 8 12 18 25 33 39 45 49 50

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 42

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 12 17 22 29 34 39 44 47 50

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 40

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 8 13 19 26 32 37 41 45 46 49 50

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 40

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 5 8 13 18 25 31 36 40 44 47 49 50

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 43

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 9 13 18 22 21 24 28 33 39 44 47 50

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 20 26 33 38 42 46 48 50

*** PROBLEMA NO. = 7 ***

FILE0 TESTE50 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS UFSC

CUSTO MINIMO DO GRAFO = 67

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 11 16 22 29 36 42 46 49 50

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 55

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 8 11 19 26 34 41 45 44 48 50

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 43

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 3 5 9 15 21 28 34 39 42 44 50

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 10 13 18 27 28 33 38 42 46 48 50

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 45

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 8 12 18 25 31 37 43 48 50

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 126

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 6 11 16 22 34 39 42 46 50

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 5 8 12 19 26 33 38 43 47 49 50

FILEO TESTE50 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 53

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 10 16 22 29 36 40 44 48 50

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 48

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 10 15 20 32 38 43 46 49 50

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 15 21 27 34 39 43 46 48 50

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 63

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 7 12 15 19 26 33 38 42 46 48 50

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 106

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 10 15 20 26 33 39 45 48 50

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 5 8 11 17 23 28 33 39 44 49 50

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 16 21 27 33 37 41 46 48 50

FILE0 TESTE100 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - LFSC

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 100

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 84

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 11 16 22 27 33 41 49 57 65 72 80
87 92 95 98 99 100

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 60

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 7 13 21 30 39 49 57 65 72 77 83 89 94
98 100

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 65

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 17 24 31 39 47 55 64 71 79 86 92
96 99 100

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 79

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 14 22 32 44 58 70 80 88 94 98 100

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 68

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 8 14 22 32 44 57 69 79 87 93 97 100

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 90

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 17 25 34 44 55 66 75 82 89 95 98

FILE=0 TESTE=100 SAIDA 41 VM/SP P4 PUT8602+ SLU408 CMS UFSC

100

*** PROBLEMA NO. = 7 ***

CUSTO MINIMO DO GRAFO = 79

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 5 10 17 25 34 45 57 68 77 85 92 97 100

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 98

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 11 17 25 34 44 56 67 76 85 92 96 99
100

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 90

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 8 13 21 31 43 54 65 74 82 89 94 98 100

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 100

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 18 26 36 48 59 69 77 84 89 94 98
100

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 84

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 9 15 23 33 45 58 71 81 88 94 98 100

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 101

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 18 24 31 40 50 60 70 77 83 89 94
98 100

*** PROBLEMA NO. = 13 ***

FILEO TESTE100 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 85

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 11 18 27 38 50 62 73 82 89 94 98 100

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 85

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 11 18 26 34 43 52 62 70 78 84 90 95
100

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 101

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 7 14 21 30 40 52 64 73 82 89 95 100

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 96

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 13 19 27 36 49 62 72 80 86 92 97 100

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 99

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 9 15 22 30 39 49 66 75 82 88 94 98 100

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 109

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 6 12 20 30 40 53 65 76 84 91 96 99 100

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 94

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 8 14 21 30 40 52 64 74 82 89 88 97 100

FILEO TESTE100 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS UFSC

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 95

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 19 37 48 60 70 79 85 90 98 100

***** ALGORITMO DE DIJKSTRA *****

NUMERO DE VERTICES DO GRAFO = 200

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 151

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 9 15 23 33 45 58 73 89 106 122 137 149 161
171 180 188 194 198 200

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 149

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 10 16 23 30 40 52 65 79 94 109 123 135
146 156 166 174 181 187 193 198 200

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 164

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 12 18 26 35 45 57 70 84 99 114 128 141
153 164 174 182 188 193 197 200

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 157

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 9 16 24 32 42 52 62 74 88 101 114 129 142
153 162 170 178 186 193 198 200

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 159

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 11 18 24 33 43 55 67 82 97 113 128 142
154 165 175 184 191 196 199 200

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 139

FILED TLSTE200 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 8 13 19 33 43 53 65 79 92 103 114 127
139 152 162 171 178 184 194 200

*** PROBLEMA NO. = 7 ***

CUSTO MINIMO DO GRAFO = 149

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 8 14 22 31 41 52 63 76 87 99 111 123 134
145 156 167 176 185 191 197 200

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 101

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 4 7 11 16 23 31 40 49 60 71 83 97 110
122 133 144 153 162 163 171 179 186 191 195 198 200

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 153

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 11 16 22 30 41 53 66 82 99 116 131 145
158 169 177 183 188 193 197 200

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 137

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 16 22 30 38 48 58 71 85 100 115 129
141 151 161 170 178 184 189 193 197 199 200

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 142

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 13 21 22 32 43 56 68 78 88 100 112 122
132 144 157 169 179 187 193 197 200

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 70

CAMINHO DE MINIMO CUSTO DO GRAFO =

FILEO TESTE200 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

1 3 5 8 12 17 22 29 37 46 56 67 79 92 108
122 134 145 154 164 171 178 179 185 190 194 197 199 200

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 75

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 11 15 21 28 36 37 46 55 66 78 90 103
116 129 141 151 160 168 175 182 187 191 195 198 200

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 135

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 5 7 10 15 21 28 36 45 54 64 75 86 97
109 120 131 142 152 162 171 178 184 190 194 197 200

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 137

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 12 19 28 37 48 60 73 87 103 118 133 146
157 166 174 180 187 192 193 198 200

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 90

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 7 8 14 21 30 41 52 66 80 95 112 127 139
151 162 173 181 187 193 197 200

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 78

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 4 9 15 16 23 30 33 40 50 62 75 89 104 119
132 145 153 168 177 184 190 195 199 200

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 71

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 6 12 19 25 31 40 47 56 70 83 96 111 110

FILE0 TESTE200 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS UFSC

123 133 143 152 162 179 186 192 197 198 200

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 76

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 3 6 10 11 17 25 33 43 54 66 80 93 107 123
136 148 159 169 179 187 193 197 199 200

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 93

CAMINHO DE MINIMO CUSTO DO GRAFO =

1 2 7 13 21 30 31 42 53 65 78 93 109 124 137
148 158 167 176 182 188 194 199 200

P R O B L E M A S D E

J E Y A R A T N A M

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 10

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 9

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 5 7 10

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 13

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 8 10

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 13

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 7 8 9 10

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 7 10

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 6 9 10

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 10

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 5 6 8 10

*** PROBLEMA NO. = 7 ***

FILEO ALGO10 SAIDA A1 VM/SP R4 PUT3602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 11

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 10

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 12

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 7 9 10

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 10

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 3 5 8 9 10

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 8 9 10

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 39

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 7 10

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 8 10

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 14

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 9 10

FILE# AL0010 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 16

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 4 5 8 9 10

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 12

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 5 6 9 10

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 14

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 6 8 9 10

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 7 9 10

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 9

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 6 8 10

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 14

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 5 9 10

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 5 8 10

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 15

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 5 8 10 13 15

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 5 7 9 11 13 14 15

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 4 5 7 9 12 15

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 6 9 11 13 15

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 4 5 8 12 15

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 6 9 10 13 15

*** PROBLEMA NO. = 7 ***

FILEO ALGO15 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 14

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 9 12 15

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 2 5 9 12 15

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 3 11 15

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 10 13 15

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 11 13 15

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 8 9 11 13 15

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 9 11 14 15

FILEO ALG015 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 7 10 12 15

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 26

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 6 8 10 13 15

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 4 7 10 13 15

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 28

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 5 9 11 15

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 13

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 3 5 9 12 14 15

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 6 10 15

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 27

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 7 10 11 14 15

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 20

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 7 10 13 18 19 20

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 14 18 19 20

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 12 16 19 20

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 10 12 16 19 20

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 7 10 13 15 17 19 20

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 10 14 18 19 20

*** PROBLEMA NO. = 7 ***

FILE0 ALGO20 SAIDA A1 VM/SP R4 PUT#602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 5 7 11 14 17 18 20

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 6 11 14 17 18 20

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 7 12 16 18 20

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 7 8 11 15 18 19 20

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 25

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 8 14 17 19 20

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 6 9 13 15 18 20

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 18

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 9 12 15 18 20

FILED ALGO20 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 6 9 13 15 16 18 19 20

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 16

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 9 13 19 20

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 9 13 17 20

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 19

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 12 16 18 20

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 10 13 16 18 19 20

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 15

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 8 11 15 18 20

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 13 17 19 20

FILE0 ALGD25 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 25

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 26

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 6 9 12 13 17 21 23 25

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 33

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 10 15 21 24 25

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 9 13 17 20 22 23 25

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 7 9 15 18 23 25

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 30

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 5 9 12 16 17 20 23 25

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 39

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 10 14 19 22 25

*** PROBLEMA NO. = 7 ***

FILED ALGO25 SAIDA A1 VM/SP R4 PUT8602+ SLU408 - CMS - UFSC

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 5 8 10 14 16 18 20 23 25

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 30

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 15 21 23 25

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 9 12 15 17 20 23 24 25

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 8 10 13 16 19 23 25

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 67

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 7 10 12 15 19 21 24 25

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 7 12 15 17 20 22 24 25

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 13 17 18 22 25

FILED ALGO25 SAIDA A1 VM/SP R4 PUT8602+ SLL408 GMS UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 61

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 10 14 19 23 25

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 69

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 7 11 16 20 23 25

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 12 15 17 21 23 24 25

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 43

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 8 14 21 24 25

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 61

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 10 17 19 22 25

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 33

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 5 6 10 14 19 22 24 25

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 30

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 12 14 18 21 23 25

FILEO ALGO30 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 30

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 27

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 12 16 20 24 27 28 29 30

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 95

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 14 17 20 23 24 28 30

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 5 7 10 12 16 20 24 27 30

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 45

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 14 19 24 27 29 30

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 16 21 23 27 29 30

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 6 9 12 18 20 22 24 27 29 30

*** PROBLEMA NO. = 7 ***

FILEO ALGO30 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 37

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 6 9 14 18 19 20 24 27 29 30

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 27

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 8 12 19 24 27 28 30

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 72

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 10 17 22 24 27 30

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 61

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 12 16 22 24 27 29 30

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 35

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 13 19 23 26 29 30

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 9 12 17 21 23 29 30

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 10 14 19 23 26 30

FILED ALGO30 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 20

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 10 13 16 23 24 28 30

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 6 11 15 20 25 28 30

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 11 16 19 27 30

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 9 13 18 24 23 27 29 30

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 8 12 16 20 25 24 28 30

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 23

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 17 20 25 27 28 30

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 21

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 13 18 23 26 30

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 35

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 4 8 14 20 26 30 33 35

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 8 13 19 25 29 34 35

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 29

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 7 13 19 24 29 33 35

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 26

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 6 10 13 19 23 27 32 33 35

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 22

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 8 12 13 19 23 24 29 30 33 35

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 24

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 7 12 19 26 30 34 35

*** PROBLEMA NO. = 7 ***

FILEO ALG053 SAIDA A1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 34

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 5 9 15 22 28 32 35

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 17

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 6 11 20 28 32 35

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 2 4 8 13 19 24 28 31 33 35

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 33

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 4 8 15 22 27 32 34 35

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 7 11 18 25 30 35

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 32

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 3 6 9 17 22 27 31 35

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 38

CAMINHO DE MINIMO CUSTO DO GRAFO 0
1 4 3 6 10 15 22 27 31 35

FILE0 ALG053 SAIDA A1 VM/SP R4 PUT8602+ SLU408 - CMS - LFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 34

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 5 8 13 14 22 25 31 33 34 35

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 48

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 15 20 25 29 33 35

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 31

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 6 9 15 21 27 31 33 35

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 106

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 7 13 20 26 31 34 35

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 41

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 9 15 21 26 30 33 35

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 37

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 15 22 28 32 35

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 32

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 20 25 29 33 35

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 40

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 14 20 25 29 34 38 40

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 53

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 8 11 15 20 25 29 33 36 39 40

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 44

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 9 13 19 25 30 34 38 40

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 8 13 22 29 34 37 39 40

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 9 13 19 26 31 35 38 40

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 5 11 16 25 31 35 37 38 40

*** PROBLEMA NO. = 7 ***

CUSTO MINIMO DO GRAFO = 45

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 10 13 17 22 28 32 36 40

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 50

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 14 19 24 29 33 36 39 40

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 88

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 9 14 17 22 28 36 38 40

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 59

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 7 11 16 21 26 31 35 38 40

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 9 14 19 23 31 34 38 39 40

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 9 14 19 24 29 33 37 40

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 51

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 20 26 31 34 38 40

FILEO ALGD40 SAIDA A1 VM/SP R4 PUT8602+ SEU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 12 16 22 28 33 36 39 40

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 50

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 19 23 28 32 35 37 39 40

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 16 22 27 37 35 38 40

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 5 9 13 18 23 28 35 38 40

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 59

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 5 11 15 21 27 33 39 40

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 53

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 10 14 19 26 31 35 38 40

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 6 9 12 17 22 28 33 36 38 40

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 45

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 58

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 14 20 32 36 40 43 45

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 8 12 16 20 24 28 33 37 41 43 44 45

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 68

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 5 13 24 30 35 39 42 44 45

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 73

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 7 10 15 27 33 38 42 45

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 9 13 18 23 29 35 39 42 45

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 64

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 7 11 15 20 30 34 38 41 45

*** PROBLEMA NO. = 7 ***

FILED ALG045 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 60

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 7 12 16 22 29 34 39 42 44 45

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 52

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 17 22 30 35 40 43 45

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 44

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 9 13 17 23 28 34 39 42 44 45

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 66

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 12 17 23 31 36 40 45

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 13 19 24 31 36 40 43 45

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 60

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 7 11 15 20 25 31 37 41 44 45

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 49

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 10 15 20 26 33 38 41 44 45

FILED ALGO45 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 62

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 19 23 27 31 34 43 44 45

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 54

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 5 8 13 18 23 28 34 38 41 44 45

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 59

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 12 16 21 27 32 36 39 41 43 45

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 68

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 9 12 18 23 34 38 41 43 45

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 81

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 17 21 26 31 36 42 44 45

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 62

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 4 6 9 14 18 25 31 37 41 44 45

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 66

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 7 16 22 33 37 40 41 44 45

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 50

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 37

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 8 12 18 25 33 39 45 49 50

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 42

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 12 17 22 29 34 39 44 47 50

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 40

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 8 13 19 26 32 37 41 42 46 49 50

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 40

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 5 8 13 18 25 31 37 41 45 47 49 50

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 43

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 9 13 18 22 21 24 28 33 39 44 47 50

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 20 26 33 38 42 46 49 50

*** PROBLEMA NO. = 7 ***

FILEO ALG050 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CUSTO MINIMO DO GRAFO = 67

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 11 16 22 29 36 42 46 49 50

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 55

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 8 11 19 26 34 41 45 44 48 50

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 43

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 3 5 9 15 21 28 34 39 42 44 50

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 46

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 10 13 18 22 28 33 38 42 46 48 50

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 45

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 8 12 18 25 31 37 43 47 50

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 126

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 6 11 16 22 34 39 42 46 50

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 5 8 14 21 28 35 41 44 47 49 50

FILE0 ALG050 SAIDA A1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 53

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 10 16 22 29 36 40 44 48 50

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 48

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 10 15 20 32 38 43 46 49 50

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 56

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 15 21 27 34 39 44 47 49 50

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 63

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 5 7 12 15 19 26 32 38 42 46 48 50

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 106

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 10 15 20 26 33 39 45 48 50

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 5 8 11 17 23 28 33 39 44 49 50

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 57

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 16 21 27 33 37 41 46 49 50

***** ALGORITMO DE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 100

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 84

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 11 16 22 27 33 41 49 57 65 72 80
87 92 95 97 99 100

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 60

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 7 13 21 30 39 49 57 65 72 77 83 89 94
98 100

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 65

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 17 24 31 39 47 55 64 71 79 86 92
96 99 100

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 79

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 13 21 31 43 57 69 79 87 93 97 100

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 68

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 8 14 22 32 44 57 69 79 87 93 97 100

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 90

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 17 25 34 44 55 66 75 82 89 95 98

FILE0 ALG0100 SAIDA C1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

100

*** PROBLEMA NO. = 7 ***

CUSTO MINIMO DO GRAFO = 79

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 5 10 17 25 34 45 57 68 77 85 92 97 100

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 98

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 11 17 25 34 44 56 67 76 85 92 96 99
100

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 90

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 8 13 21 31 43 54 65 74 82 89 94 98 100

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 100

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 12 18 26 36 48 59 69 76 83 89 94 98
100

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 84

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 9 15 23 33 45 58 71 81 88 94 98 100

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 101

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 12 18 24 31 40 50 60 70 77 83 89 94
98 100

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 85

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 11 18 27 38 50 62 73 82 89 94 98 100

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 85

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 6 11 18 26 34 43 52 62 70 78 84 90 95
100

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 101

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 7 14 21 30 40 52 64 73 82 89 95 100

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 96

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 8 9 15 21 28 37 49 62 72 80 86 92 97
100

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 99

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 4 9 15 22 30 39 49 66 75 82 88 94 98 100

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 109

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 6 12 20 30 40 53 65 76 84 91 96 98 100

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 94

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 8 14 21 30 40 52 64 74 82 89 88 97 100

FILEO ALGO103 SAIDA C1 VM/SP R4 PUT8602+ SLU408 -- CMS - UFSC

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 95

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 12 19 37 48 60 70 79 85 91 95 100

***** ALGORITMO CE S. JEYARATNAM *****

NUMERO DE VERTICES DO GRAFO = 200

*** PROBLEMA NO. = 1 ***

CUSTO MINIMO DO GRAFO = 151

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	4	9	15	23	33	45	58	73	89	106	122	137	149	161
171	180	188	194	198	200									

*** PROBLEMA NO. = 2 ***

CUSTO MINIMO DO GRAFO = 149

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	2	5	10	16	23	30	40	52	65	79	94	109	123	135
146	156	166	174	182	188	193	198	200						

*** PROBLEMA NO. = 3 ***

CUSTO MINIMO DO GRAFO = 164

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	3	7	12	18	26	35	45	57	70	84	99	114	128	141
153	164	174	182	188	193	197	200							

*** PROBLEMA NO. = 4 ***

CUSTO MINIMO DO GRAFO = 157

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	4	9	16	24	32	42	52	62	74	88	101	114	129	142
153	162	171	179	186	193	198	200							

*** PROBLEMA NO. = 5 ***

CUSTO MINIMO DO GRAFO = 159

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	2	5	11	18	24	33	43	55	67	82	97	113	128	142
154	165	175	184	191	196	199	200							

*** PROBLEMA NO. = 6 ***

CUSTO MINIMO DO GRAFO = 139

FILEO ALGO200 SAIDA C1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 8 13 19 33 43 53 65 79 92 103 114 127
139 152 162 171 178 184 194 200

*** PROBLEMA NO. = 7 ***

CUSTO MINIMO DO GRAFO = 149

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 8 14 22 31 41 52 63 76 87 99 111 123 134
145 156 157 176 185 191 197 200

*** PROBLEMA NO. = 8 ***

CUSTO MINIMO DO GRAFO = 101

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 4 7 11 16 23 31 40 49 60 71 83 97 110
122 133 144 153 162 163 171 179 186 191 195 198 200

*** PROBLEMA NO. = 9 ***

CUSTO MINIMO DO GRAFO = 153

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 11 16 22 30 41 53 66 82 99 116 131 145
158 169 177 183 188 193 197 200

*** PROBLEMA NO. = 10 ***

CUSTO MINIMO DO GRAFO = 137

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 16 22 30 38 48 58 71 85 100 115 129
141 151 151 170 178 184 189 193 197 199 200

*** PROBLEMA NO. = 11 ***

CUSTO MINIMO DO GRAFO = 142

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 7 13 21 22 32 43 56 68 78 88 100 112 122
132 144 157 169 179 187 193 197 200

*** PROBLEMA NO. = 12 ***

CUSTO MINIMO DO GRAFO = 70

CAMINHO DE MINIMO CUSTO DO GRAFO 0

FILE0 ALGO200 SAIDA C1 VM/SP R4 PUT8602+ SLL408 CMS - UFSC

1	3	5	8	12	17	22	29	37	46	56	67	79	92	108
122	134	145	154	164	171	178	179	185	190	194	197	199	200	

*** PROBLEMA NO. = 13 ***

CUSTO MINIMO DO GRAFO = 75

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	3	7	11	15	21	28	36	37	46	55	66	78	90	103
116	129	141	151	160	168	175	182	187	191	195	199	200		

*** PROBLEMA NO. = 14 ***

CUSTO MINIMO DO GRAFO = 135

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	2	5	7	10	15	21	28	36	45	54	64	75	86	97
109	120	131	142	152	162	171	178	184	190	194	197	200		

*** PROBLEMA NO. = 15 ***

CUSTO MINIMO DO GRAFO = 137

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	2	6	12	19	28	37	48	61	75	89	104	119	133	146
157	166	174	180	187	192	193	198	200						

*** PROBLEMA NO. = 16 ***

CUSTO MINIMO DO GRAFO = 90

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	3	7	8	14	21	30	41	52	66	80	95	112	127	139
151	162	173	181	187	193	197	200							

*** PROBLEMA NO. = 17 ***

CUSTO MINIMO DO GRAFO = 78

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	4	9	15	16	23	30	39	40	50	62	75	89	104	119
132	145	158	168	177	184	190	195	199	200					

*** PROBLEMA NO. = 18 ***

CUSTO MINIMO DO GRAFO = 71

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1	2	6	12	19	25	31	40	47	56	70	83	96	110	123
---	---	---	----	----	----	----	----	----	----	----	----	----	-----	-----

FILED ALGO200 SAIDA C1 VM/SP R4 PUT8602+ SLU408 CMS - UFSC

133 143 152 162 179 186 192 197 198 200

*** PROBLEMA NO. = 19 ***

CUSTO MINIMO DO GRAFO = 76

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 3 6 10 11 17 25 33 43 54 66 80 94 109 123
136 148 159 169 179 187 193 197 199 200

*** PROBLEMA NO. = 20 ***

CUSTO MINIMO DO GRAFO = 93

CAMINHO DE MINIMO CUSTO DO GRAFO 0

1 2 7 13 21 30 31 42 53 65 78 93 109 124 137
148 158 167 176 182 188 194 199 200
