بسم الله الرحمن الرحيم

CPAS مركز الدراسات التخطيطية والمعمارية

CENTER OF PLANNING AND ARCHITECTURE STUDIES

الدورة التدريبية الرابعة

" ادارة عمليات التشييد والبناء "

مجموعة محاضرات

د. حمدى الشيخ

لعام ١٤٠٣ – ١٩٨٢ م.

14, EL SOBKY ST., M. EL-BAKRY—HELIOPOLIS—CAIRO—EGYPT.
P.O.B. : 6 SARAY-EL-KOBBA — CABLE : ARPLANEC
TEL. : 603843 — 603397 — TELEX : 93243 CPAS UN

١٤ ش السبكى منشية البكرى – هليوبوليس – القاهرة – ج. م. ع
ص. ب : ٦ سراى القبة – تلغرافيا : اربلانك .
ت : ٦٠٣٨٤٣ – ٦٠٣٣٩٧ تلكس : CPAS UN ٩٣٢٤٣

مجموعة محاضرات

الدكتور/ حمدى الشيخ

" الجهاز الحاسب في تخطيط وتنفيذ المشروعات"

الدكتور/ حمدى الشيخ

" دراسة المسار الحرج فى المشروعات "

الدكتور/ حمدى الشيخ

بسم الله الرحمن الرحيم

فكرة عامة عن امكانيات الجهاز الحاسب
فى تخطيط وتنفيذ المشروعات واتخاذ القرارات

مقدمة :

عند ما القت امريكا القنبلة الذريه على هيروشيما وناجازاكى ، قام العلماء باجراء جميع حساباتهم بالطرق اليدوية والميكانيكية ، ولما فكر تيلر فى انتاج القنبلة الهيدروجينية كلف فريقا من العلماء بقيادة فون نيومان بتصميم اول جهاز حاسب الكترونى سمى EDVAC حتى يمكن اجراء حسابات القنبلة الهيدروجينية علية ، وهكذا بدأت الحاسبات الالكترونية كجزء مكمل لاضخم المشروعات العلمية وطبيعى ان يكون النتاج الطبيعى لهذا شيئا ضخما معقدا غالى التكاليف هو مايسمى بالحاسب الالكترونى .

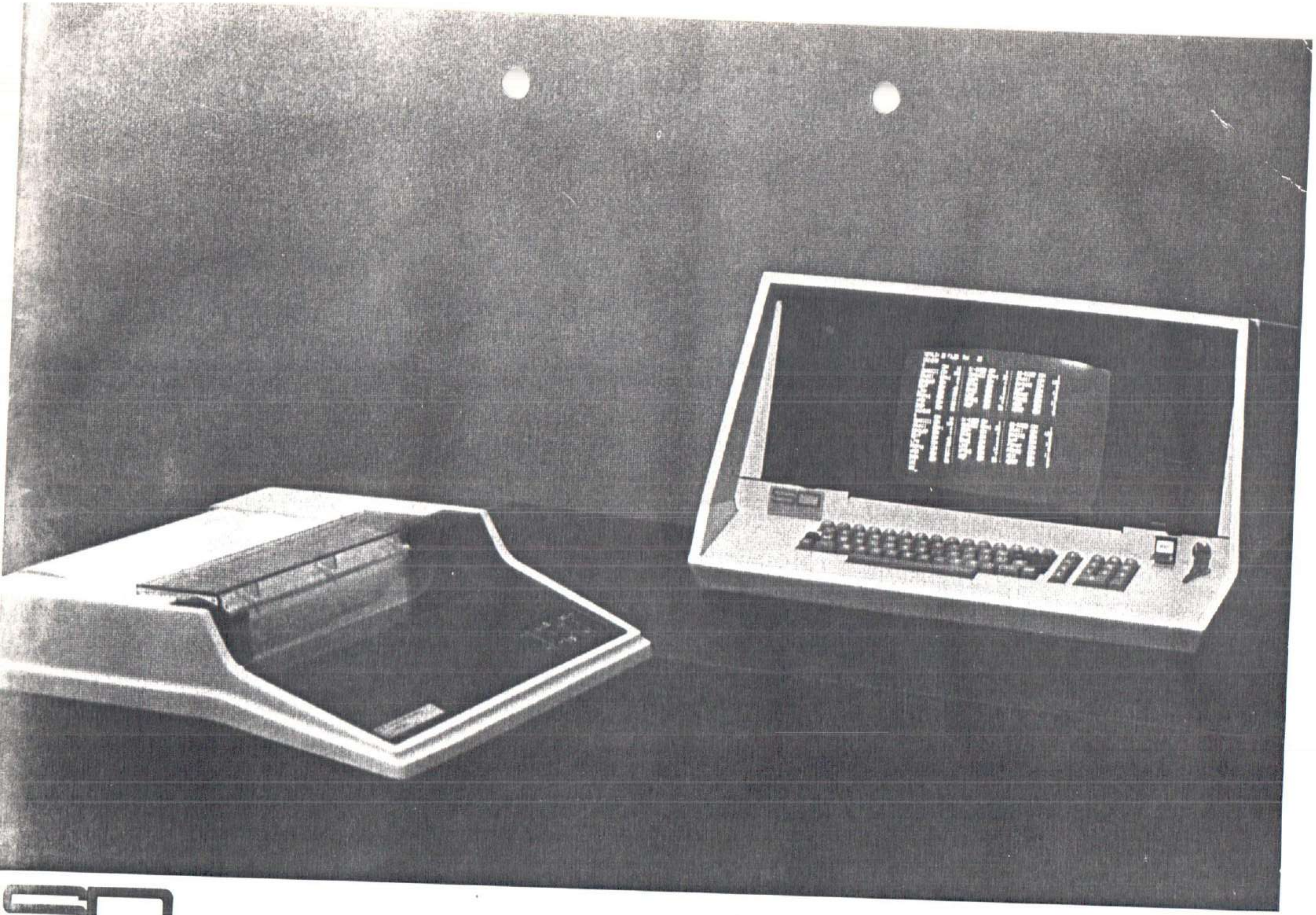
ولقد مضى زمن مقداره ثلاثون عاما منذ ذلك التاريخ قبل ان تحدث المعجزة ويتحول الحاسب من عملاق معقد كئيب الى الة مطيعة صغيرة لها نفس الامكانيات وفى متناول الانسان العادى ان يتعامل معها كما يتعامل مع جهاز التليفزيون فى منزله .. ولم تعد الحاسبات مقصورة على المتخصصين فى الالكترونيات والبرمجة والرياضيات وفجآة تصدعت اركان امبراطورية آى بى ام ومثيلاتها من الامبراطوريات ، ومن العجب العجاب ان السر وراء ذلك يكمن فى الخواص الطبيعية لمادة من مواد البناء الا وهى السليكون .

فقد استطاع فريق من علماء الصناعة اختراع مايسمى بالدوائر المتكاملة حيث يمكن تجميع مئات وآلاف الدوائر الالكترونية فى قطعة واحدة مصنوعة من السليكون لايزيد حجمها عن نصف سنتيمتر مربع وقد ترتب على ذلك انه اصبح من الممكن تصنيع اهم جزء من الحاسب فى دائرة متكاملة واحدة لاتزيد مساحتها على بوصة مربعة واحدة واصبحت الوحدة المركزية للحاسب

التى لاتشغل الا هذا الحيز البسيط لاتزيد فى تكلفتها عن عشرين دولارا بدلا من الالوف المؤلفة والاهم من ذلك انها تراجعت فبعد ان كانت تشغل عدة كبائن مليئة بالاسلاك المعقدة وتحتاج الى تبريد ووقاية وصيانة الخ ، والى حاشية من المهندسين والاخصائيين ، هذا بالاضافة الى ان درجة الاعتماد عليها زادت Reliability مئات المرات واذا حدث ونادرا مايحدث ذلك ان تعطلت اصبح ابدالها ارخص من محاولة اصلاحها •

وبالمثل حدث تطور فى جميع اجزاء الحاسبات واصبحنا منذ ١٩٧٦ نتعامل مع مايسمى بالحاسبات الشخصية Personal Computers التى لايزيد حجمها عن جهاز التليفزيون ولايزيد ثمنها عن ثمن سيارة والتى يمكن حتى للاطفال فى مدارسهم وبيوتهم التعامل معها وهنا احذركم من كهنة الحاسبات الذين تربو فى مدرسة الحاسبات الضخمة والتى لازالت مع الاسف موجودة من انتقاد للحاسبات الشخصية بحجة ضعف امكانياتها عن الحاسبات الكبيرة • كلا فالامكانيات متقاربة ولاتبرر زيادة السرعة الى الضعف او يزداد الثمن مئةمرة والاهم من ذلك ان الحاسبات الشخصية طبيعة سهلة مصممة لاستخدام الانسان العادى ولن يكون بعيدا ذلك اليوم عندما تصبح هذه الحاسبات جزءا اساسيا ضمن المعدات المنزلية حيث يستخدمها الاطفال فى استذكار دروسهم وتستخدمها ربة المنزل فى اعداد وجبات الطعام ويستخدمها رب المنزل فى ترتيب مواعيده واعداد خطاباتة ويستخدمها المهندس المعمارى او الانشائى فى ابداع مشاريعـــــة •

ولن اخوض معكم ولااحب ان يكون ذلك منهجى فى خضم من المصطلحات الفنية التى تعود خبراء الحاسبات فى كلامهم عنها ان يحشروها فى ثنايا الكلمات ومع ذلك فلابد من كلمة قصيرة نتناول فيها تركيب الحاسب الالكترونى حتى نستطيع استخدامة بوعى كامل لامكانياته •



SD SYSTEMS

A SUBSIDIARY OF THE MICHIGAN GENERAL CORPORATION

شكـــل (رقـــم ٥)

SD - 100/200

إنتاج ١٩٨٠
الثمن مايعادل ٢٠٠ برميل بترول
فرد واحد للتشغيل

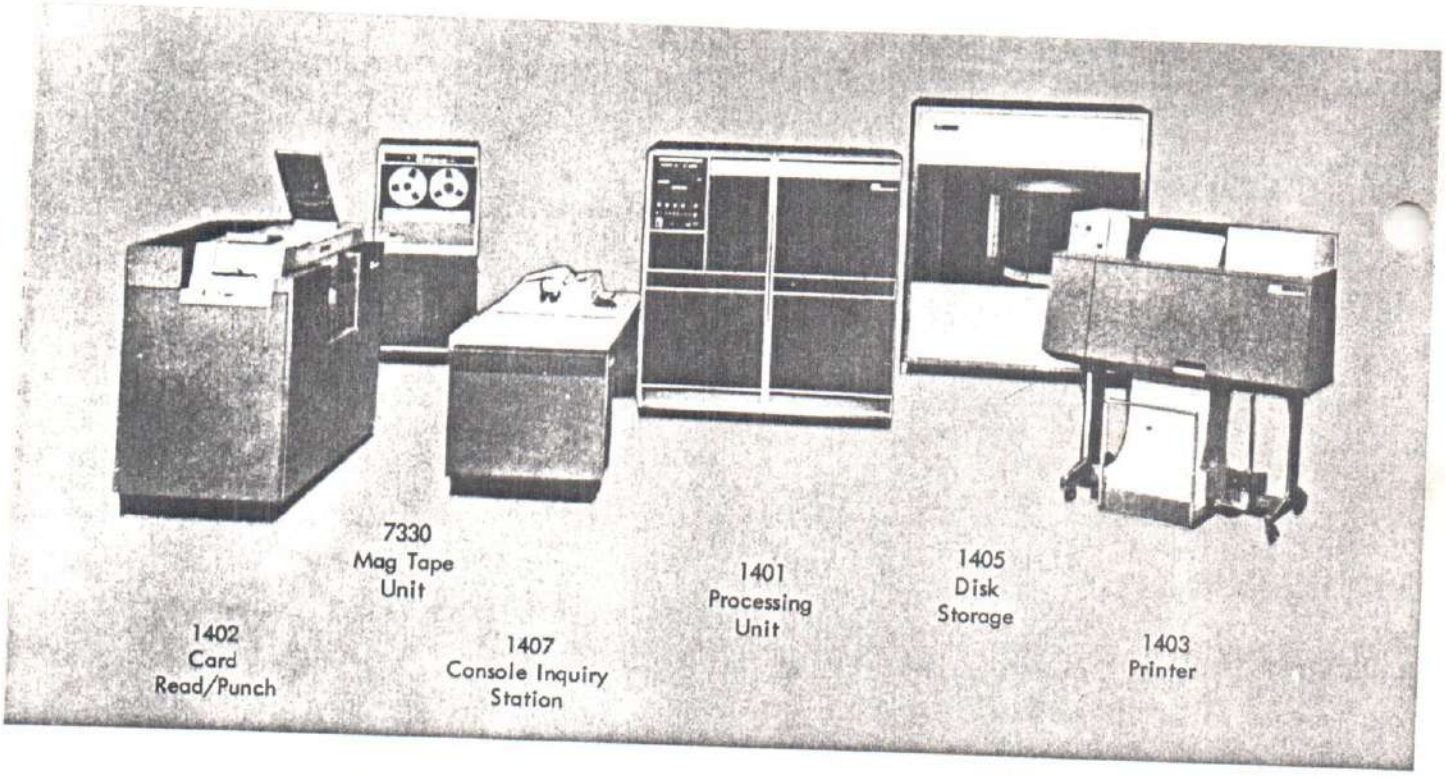


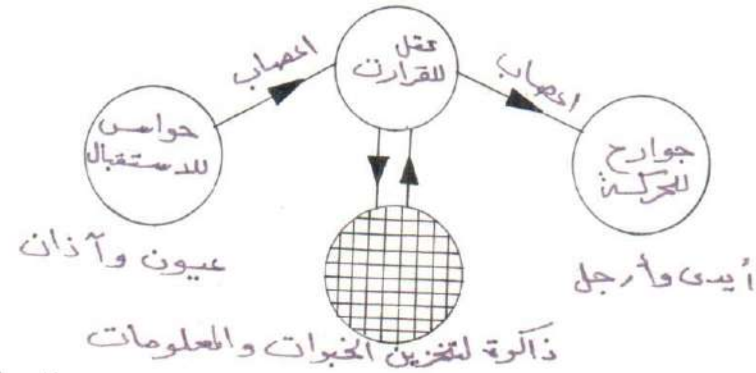
Figure 2–4. IBM 1401 Data Processing System (Photograph, IBM)

إنتاج عام ١٩٦٢
الثمن مايعادل ١٠٠ ألف برميل بترول
الجهاز الوظيفي : ١٥ موظف للتشغيل

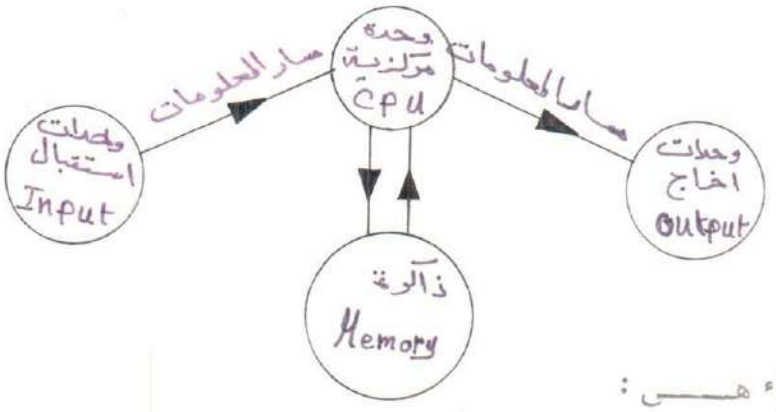
شكـــــــــل (رقـــــم ٦)

فكرة مختصرة عن الحاسب الالكترونى :

يتركب الحاسب الالكترونى مثل اى كائن مركب من عدة اجزاء تتفاعل مع بعضها لتؤدى الوظائف الخاصة بهذا الكائن فالانسان مثلا مكون من عقل ليفكر ويتخذ القرارات وذاكرة تختزن الخبرة والمعلومات وحواس تستقبل المعلومات وجوارح تقوم بالحركة وتنفيذ القرارات واذا حاولنا تمثيل ذلك بلغة المهندسيين فلن يزيد ذلك عن الشكل الاتى :



وعلى نفس النمط فان الجهاز الحاسب يتكون من اجزاء شبيهة من حيث الخواص الوظيفية .



وهذه الاجزاء هى :

١ـ الوحدة المركزية للتشغيل Central Processing Unit وقد تسمى بالوحدة الحسابية المنطقية Arithmetic Logic Unit (Alu) وهى المنوطة باجراء العمليات المنطقية والحسابية وتشغيل جميع الوحدات الاخرى .

٢ـ وحدة اووحدات الذاكرة Memory وهى المستخدمة فى تخزين المعلومات بمافيها برامج التعامل مع هذه المعلومات .

٣ـ وحدات الادخال Input Units وهى المستخدمة فى ادخال البيانات الى الحاسب او ادخال التعليمات والبرامج ٠

٤ـ وحدات الاخراج Output Units وهى المستخدمة فى اخراج النتائج المترتبة على تشغيل الحاسب وقد تكون على صورة طابع يخرج النتائج مطبوعة فى صورة مكتوبة او راسم يخرج النتائج او اشارة تتحكم فى جهاز ٠

متى يجب استخدام الحاسب الالكترونى وكيف يستخدم :

بناء على بعض الدراسات الحديثة التى اجريت على الجدوى الاقتصادية لاستخدام الحاسبات الالكترونية فى الاعمال البسيطة مثل كتابة الرسائل ورسم اللوحات الهندسية البسيطة تبين مايلى :

ـ بالنسبة لارسال الخطابات يصبح استخدام الحاسب اقتصاديا اذا تكرر كتابة نفس الخطاب لاكثر من سبعة افراد ويمكن لحاسب ثمنه مع آله الطباعة حوالى ٨٠٠٠ دولار ان يقوم بعمل ثلاثة سكرتيرات٠

ـ بالنسبة للرسومات الهندسية البسيطة يصبح استخدام الحاسب اقتصاديا اذا اجرى على الرسم اكثر من ثلاثة تعديلات قبل اعتمادة ويمكن لحاسب مزود بجهاز رسم ثمنه حوالى ٨٠٠٠ دولار ان يقوم بعمل رسامين ٠

هذا بالاضافة ان هذا النوع من التطبيقات اما فى التطبيقات التقليدية مثل حسابات المخازن وضبطها فانه قد ثبت انه اذا تعدى عدد الاصناف الموجودة فى المخزن ٧٠٠ صنف مع حركة يومية تزيد عن ٣٥ صنف فان استخدام حاسب صغير لايزيد ثمنه عن ٨٠٠٠ دولار يصبح اقتصاديا ويمكنة توفير عمل ثلاثة من العاملين ٠

اما فى التطبيقات التى تحتاج الى اجراء عمليات حسابية معقدة مثل حسابات الخرسانة

وغيرها من التطبيقات العملية فان استخدام الحاسبات يوفر الكثير من الوقت والجهد وخصوصا فى الحالات ذات البدائل المتشابهة والمتكررة •

ماهى طريقة المسار الحرج Critical Path Method (C.P.M.)

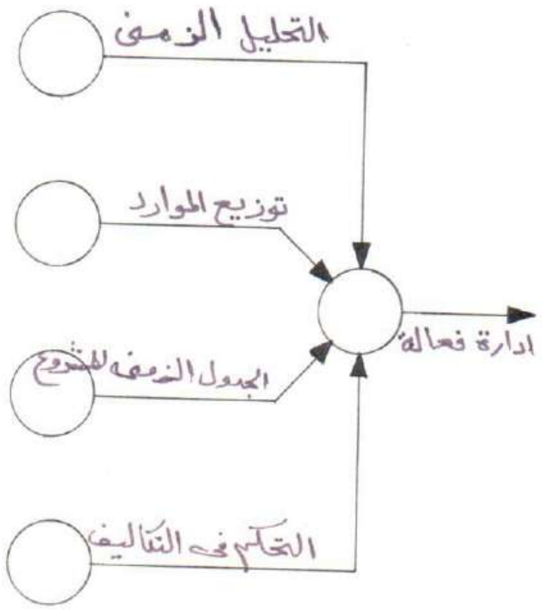
عندما كانت شركة لوكهيد تقوم بتصنيع طائراتها كانت تعانى دائما من عدم مقدرتها على الالتزام بتنفيذ العقود المبرمة مع الحكومة الامريكية طبقا للجداول الزمنية فاستعانت بمجموعة من علماء بحوث العمليات الذين وجدوا ان عدد الاجزاء المطلوبة لتصنيع الطائرة ضخم جدا يتجاوز عشرات الالاف وان تنظيم عملية الامداد بهذه الاجزاء وتصنيعها بحيث يمكن اتمام المنتج فى وقت معروف مقدما يحتاج الى طريقة جديدة وقد اصطلح على تسمية هذه الطريقة بطريقة المسار الحرج Critical Path Method (C.P.M.) او طريقة بيرت اختصار للجملة الانجليزية (PERT) Program Evaluation & Review Technique

ومنذ ذلك الحين اصبحت هذه الطريقة من اهم طرق التخطيط والتحكم فى المشروعات•

ولنأخذ مثالا بسيطا :

لنفترض انه مطلوب تخطيط مشروع كبير وليكن انشاء محطة كهربائية من المعلوم ان الخطوات التنفيذية للمشروع قد تكون كمايلى :

- اجراء الاعمال المساحية •
- اجراء عمليات الحفر •
- اجراء عمليات الخرسانة •
- التعاقد على توريد الماكينات•
- تخزين الماكينات والمعدات•



- اتمام المبانى •
- تركيب الماكينات •
- تجربة الماكينات •
- تسليم المحطة •

ومن المعلوم ان هذه العمليات يجب ان تجرى بترتيب زمنى معين فمثلا لايمكن اجراء عمليات الحفر او صب الخرسانة قبل اجراء العمليات المساحية كما لايمكن تركيب الماكينات قبل اتمام القواعد الخرسانية كما ان عملية تستغرق وقتا معينا يمكن تقديره مقدما ، وقد يتأخر تنفيذ عملية من العمليات عن الوقت المقيم لها ، فما هو تأثير ذلك على باقى العمليات فمثلا قد يتأخر تخزين المعدات ولايوءثر ذلك على اتمام المبانى • وعلى ادارة المشروع ان تتولى تخطيط كل هذه العمليات مجتمعة كما تتولى مراقبة تنفيذ كل عملية على انفراد والتنسيق بين هذه العمليات فمثلا يكون الوضع مثاليا بالنسبة لمدير المشروع لو ان كل المعدات والماكينات كانت مشونة بموقع المشروع قبل ان يبدأ ، بينما يتم المدير المالى للمشروع بان يوءخر عملية استلام المعدات الى آخر نقطة حرجة يمكنه تاخير هذه العملية اليها ، حيث يوفر بذلك الفوائد على راس المال المستثمر فى شراء الماكينات لعام او اكثر •

ملحوظة جانبية :

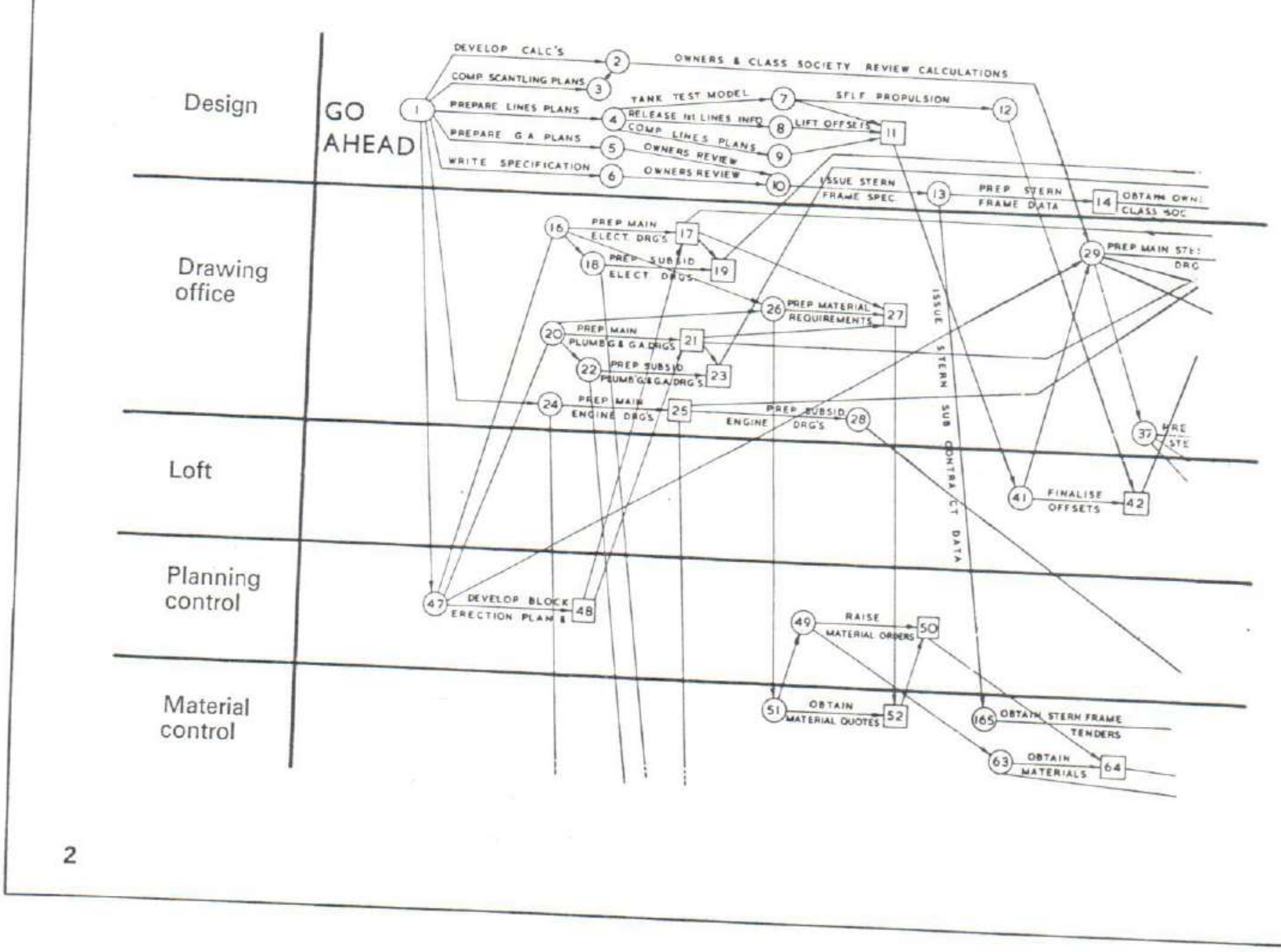
فى الدول النامية كثيرا ماتشون الماكينات ولاينفذ المشروع من اصله وهذا موءشر على التى يتمتع بها المديرون التنفيذيون • كل هذه الاشياء يمكن معالجتها بطريقة المسار الحرج •

التخطيط بالشبكات:

تتفق كل الطرق المستخدمة فى التخطيط فى استخدام الاسهم Arrow Diagram لبيان الترتيب المنطقى للنشاطات التى يجب ان تتم قبل اتمام المشروع بحيث تتمكن الادارة باستخدام هذه الطرق من التوصل الى تحكم افضل فى المشروعات وخصوصا فى مجالات تخصيص


EXCAVATE & CONCRETE
ERECT STEEL WORK
111
112
LAY ROOF DECKING & GLAZE ROOF LIGHTS (MAIN BUILDING)
113
114
117
LAY MAIN R.C. SHELL SLAB (MAIN BUILDING)
119
118
CONCRETE STEEL WORK IN EAST ANNEXE
123
124
CONCRETE STEELWORK IN WEST ANNEXE
130
122
FIT PRE-CAST ROOF SLAB (E ANNEXE)
125
132
FIT PRE-CAST ROOF SLAB W ANNEXE
135
126
EXCAVATE/LAY E ANNEXE DRAINS
129
133
EXCAVATE/LAY W ANNEXE DRAINS
136
127
LAY E ANNEXE R.C. SITE SLAB
128
134
LAY W ANNEXE R.C. SITE SLAB
137
142
BUILD EXTERNAL BRICK WALLS (W ANNEXE)
143
144
FIT RAINWATER PIPES / MAKE ROOF WATERTIGHT (W ANNEXE)
145
131
BUILD EXTERNAL BRICK WALLS (E ANNEXE)
139
138
FIT RAINWATER PIPES / MAKE ROOF WATERTIGHT (E ANNEXE)
148
146
BUILD EXTERNAL BRICK WALL (MAIN BUILDING FROM STEW)
147
149
CLAD WALL (MAIN BUILDING)
150
115
FIT HEATING/ELECT SERVICES IN ROOF (MAIN BUILDING)
116
120
BUILD INTERNAL FIRE WALL (MAIN BUILDING)
121
153
BUILD INTERNAL BRICK WALLS (W ANNEXE)
154
INSTALL SERVICES (W ANNEXE)
155
140
BUILD INTERNAL BRICK WALLS (E ANNEXE)
151
INSTALL SERVICES (E ANNEXE)
152

1


Design
Drawing office
Loft
Planning control
Material control
GO AHEAD
1
DEVELOP CALC'S
2
OWNERS & CLASS SOCIETY REVIEW CALCULATIONS
COMP SCANTLING PLANS
3
PREPARE LINES PLANS
4
TANK TEST MODEL
7
SELF PROPULSION
12
RELEASE 1st LINES INFO
8
LIFT OFFSETS
11
COMP LINES PLANS
9
PREPARE G A PLANS
5
OWNERS REVIEW
WRITE SPECIFICATION
6
OWNERS REVIEW
10
ISSUE STERN FRAME SPEC
13
PREP STERN FRAME DATA
14
OBTAIN OWN CLASS SOC
16
PREP MAIN ELECT DRG'S
17
18
PREP SUBSID ELECT DRGS
19
26
PREP MATERIAL REQUIREMENTS
27
20
PREP MAIN PLUMB'G & G A DRG'S
21
22
PREP SUBSID PLUMB'G & G.A DRG'S
23
24
PREP MAIN ENGINE DRG'S
25
PREP SUBSID ENGINE DRG'S
28
29
PREP MAIN STE DRG
37
PRE STE
ISSUE STERN SUB CONTRACT DATA
41
FINALISE OFFSETS
42
47
DEVELOP BLOCK ERECTION PLAN
48
49
RAISE MATERIAL ORDERS
50
51
OBTAIN MATERIAL QUOTES
52
165
OBTAIN STERN FRAME TENDERS
63
OBTAIN MATERIALS
64

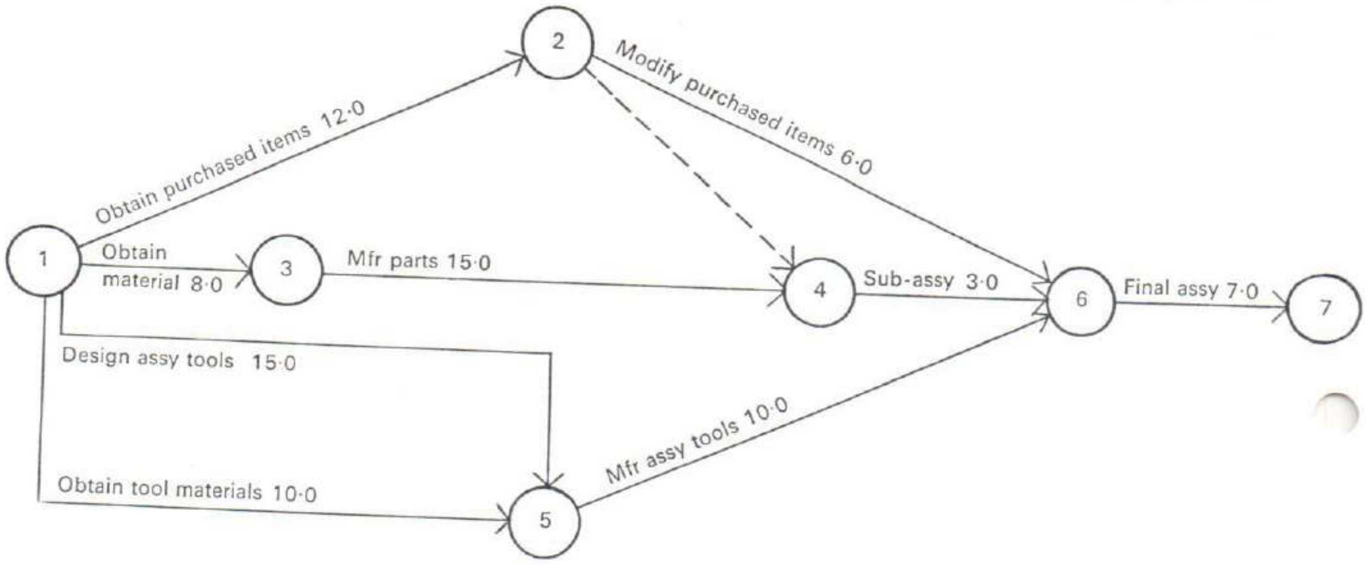
2

الموارد ، والتحكم فى التكاليف واعداد عدة جداول زمنية بديلة ان اعداد الجداول الزمنية للمشروعات الصغيرة قد يمكن القيام بهيدويا اما فى المشروعات التى تحتوى على اكثر من خمسين عملا منفصلا فان الحساب اليدوى يصبح اكثر تكلفة وفيه مضيعة للجهد وللوقت وهنا لامفر من استخدام الحاسب ، وعندما كانت الحاسبات غالية الثمن لم تكن الادارة تستخدم هذه الطرق فى تخطيط المشروعات الا فى المشروعات الضخمة مثل اطلاق الصواريخ وبناء الطائرات والكبارى والسفن وانشاء الطرق اما الان فيمكن استخدامها فى مشاريع اقل من هذه بكثير مثل انشاء عنبر جديد فى مصنع نقل ورشة تعديل مشروع قائم او تغيير عملية من عمليات التصنيع ، او حتى القيام بحملة دعاية اوالسيطرةعلى عمليات مقاولات الباطن ٠٠ الخ وسنرى ذلك فى الامثلة التى سنوردها فى التطبيق العملى على الحاسب ٠

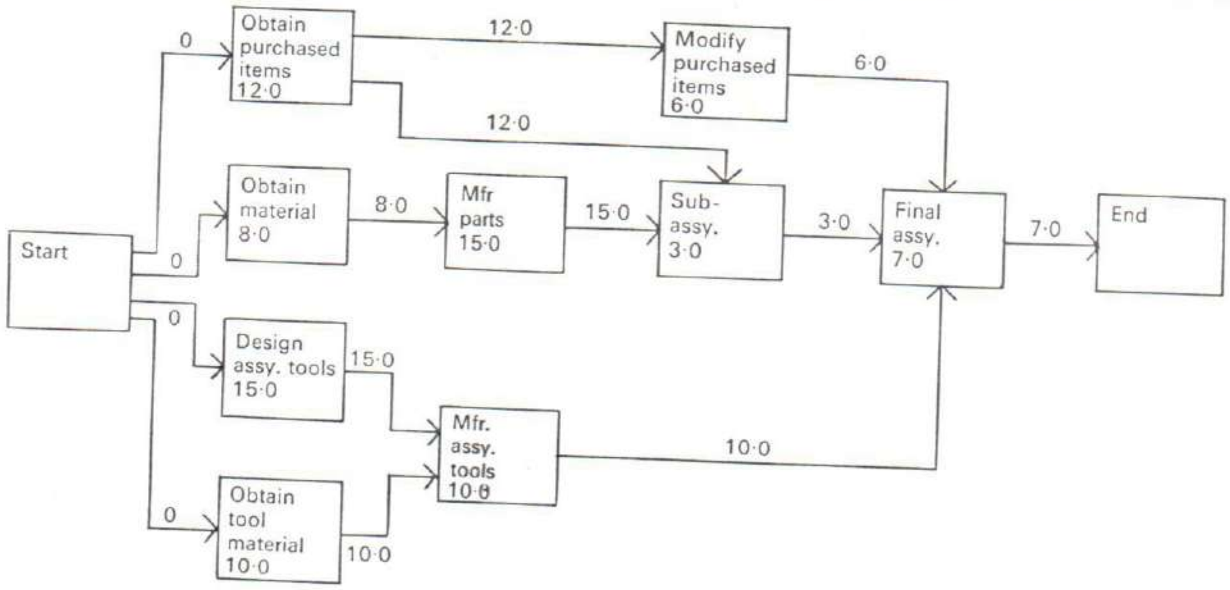
وفى شبكة بيرت تمثل الوظائف او النشاطات المنفردة بخطوط ذات اسهم تبتدىء وتنتهى فى نقط معينة ٠ هذه النقط تسمى حوادث وعادة ماتمثل على الشبكة بدائرة او مربع وعلية فان كل النشاطات والحوادث تدخل فى الشبكة ، وطبعا يختلف مستوى عمق التفاصيل الموضحة بالشبكة من مشروع الى مشروع ويوضح الشكل (١) جزءا من شبكة تنفيذ احد مشاريع البناء ٠

ويقوم باعداد هذة الشبكة بواسطة مجموعة من العاملين القائمين بتنفيذ المشروع ممايساعد على توضيح اى غموض او سوء فهم عند بدأ التنفيذ ، وقد تقسم الشبكة افقيا حسب المسئوليات كما هو موضح فى شكل (٢) الذى يمثل جزءا من شبكة لتنفيذ مشروع بناء سفينة ، مقسما حسب الادارات المسئولة عن المشروع ٠

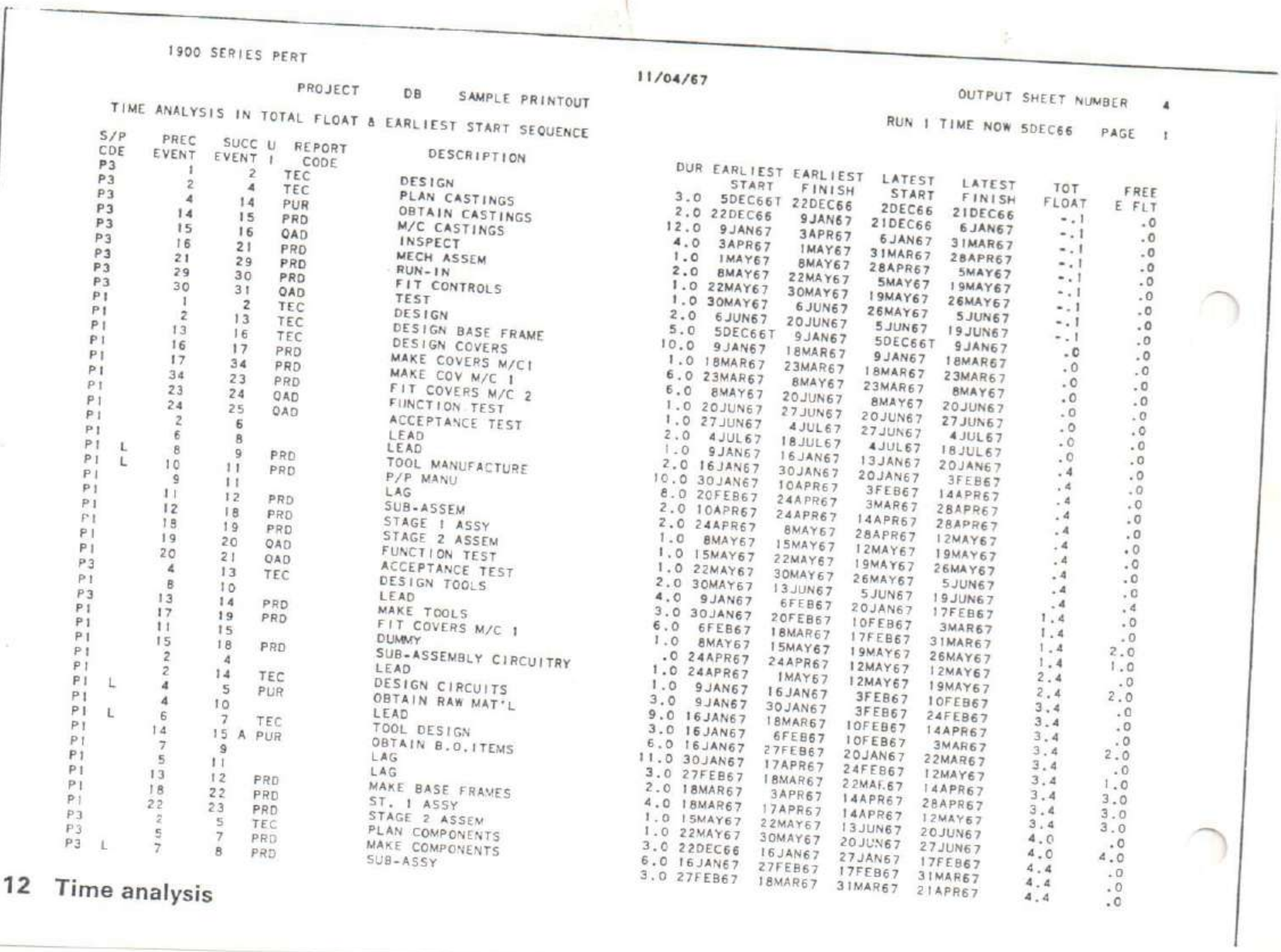
ويمثل شكل (٣) شبكة بيرت لمشروع تجميع آلات ومنه يتضح المسار الحرج المتمثل فى الحوادث ١ ، ٣ ، ٤ ، ٦ ، ٧ وتبين الارقام الموضحة على الشكل الزمن اللازم للقيام بكل نشاط من النشاطات بالاسابيع وتوضح هذه الشبكة ان الزمن الكلى اللازم لتنفيذ المشروع هو ثلاثة وثلاثون اسبوعا ٠



3 PERT Network



4 Precedence Diagram



```
        1900 SERIES PERT
                                                                    11/04/67
                         PROJECT     DB    SAMPLE PRINTOUT                                                  OUTPUT SHEET NUMBER    4
     TIME ANALYSIS IN TOTAL FLOAT & EARLIEST START SEQUENCE                                     RUN 1 TIME NOW 5DEC66   PAGE   1
  S/P     PREC   SUCC U  REPORT
  CDE     EVENT  EVENT I   CODE        DESCRIPTION                 DUR EARLIEST EARLIEST   LATEST    LATEST      TOT     FREE
                                                                          START   FINISH     START    FINISH    FLOAT    E FLT
  P3          1      2   TEC        DESIGN                          3.0  5DEC66T  22DEC66    2DEC66   21DEC66      -.1      .0
  P3          2      4   TEC        PLAN CASTINGS                   2.0  22DEC66   9JAN67   21DEC66    6JAN67      -.1      .0
  P3          4     14   PUR        OBTAIN CASTINGS                12.0   9JAN67   3APR67    6JAN67   31MAR67      -.1      .0
  P3         14     15   PRD        M/C CASTINGS                    4.0   3APR67   1MAY67   31MAR67   28APR67      -.1      .0
  P3         15     16   QAD        INSPECT                         1.0   1MAY67   8MAY67   28APR67    5MAY67      -.1      .0
  P3         16     21   PRD        MECH ASSEM                      2.0   8MAY67  22MAY67    5MAY67   19MAY67      -.1      .0
  P3         21     29   PRD        RUN-IN                          1.0  22MAY67  30MAY67   19MAY67   26MAY67      -.1      .0
  P3         29     30   PRD        FIT CONTROLS                    1.0  30MAY67   6JUN67   26MAY67    5JUN67      -.1      .0
  P3         30     31   QAD        TEST                            2.0   6JUN67  20JUN67    5JUN67   19JUN67      -.1      .0
  P1          1      2   TEC        DESIGN                          5.0  5DEC66T   9JAN67   5DEC66T    9JAN67       .0      .0
  P1          2     13   TEC        DESIGN BASE FRAME              10.0   9JAN67  18MAR67    9JAN67   18MAR67       .0      .0
  P1         13     16   TEC        DESIGN COVERS                   1.0  18MAR67  23MAR67   18MAR67   23MAR67       .0      .0
  P1         16     17   PRD        MAKE COVERS M/C1                6.0  23MAR67   8MAY67   23MAR67    8MAY67       .0      .0
  P1         17     34   PRD        MAKE COV M/C 1                  6.0   8MAY67  20JUN67    8MAY67   20JUN67       .0      .0
  P1         34     23   PRD        FIT COVERS M/C 2                1.0  20JUN67  27JUN67   20JUN67   27JUN67       .0      .0
  P1         23     24   QAD        FUNCTION TEST                   1.0  27JUN67   4JUL67   27JUN67    4JUL67       .0      .0
  P1         24     25   QAD        ACCEPTANCE TEST                 2.0   4JUL67  18JUL67    4JUL67   18JUL67       .0      .0
  P1          2      6              LEAD                            1.0   9JAN67  16JAN67   13JAN67   20JAN67       .4      .0
  P1          6      8              LEAD                            2.0  16JAN67  30JAN67   20JAN67    3FEB67       .4      .0
  P1  L       8      9   PRD        TOOL MANUFACTURE               10.0  30JAN67  10APR67    3FEB67   14APR67       .4      .0
  P1  L      10     11   PRD        P/P MANU                        8.0  20FEB67  24APR67    3MAR67   28APR67       .4      .0
  P1          9     11              LAG                             2.0  10APR67  24APR67   14APR67   28APR67       .4      .0
  P1         11     12   PRD        SUB-ASSEM                       2.0  24APR67   8MAY67   28APR67   12MAY67       .4      .0
  P1         12     18   PRD        STAGE 1 ASSY                    1.0   8MAY67  15MAY67   12MAY67   19MAY67       .4      .0
  P1         18     19   PRD        STAGE 2 ASSEM                   1.0  15MAY67  22MAY67   19MAY67   26MAY67       .4      .0
  P1         19     20   QAD        FUNCTION TEST                   1.0  22MAY67  30MAY67   26MAY67    5JUN67       .4      .0
  P1         20     21   QAD        ACCEPTANCE TEST                 2.0  30MAY67  13JUN67    5JUN67   19JUN67       .4      .4
  P3          4     13   TEC        DESIGN TOOLS                    4.0   9JAN67   6FEB67   20JAN67   17FEB67      1.4      .0
  P1          8     10              LEAD                            3.0  30JAN67  20FEB67   10FEB67    3MAR67      1.4      .0
  P3         13     14   PRD        MAKE TOOLS                      6.0   6FEB67  18MAR67   17FEB67   31MAR67      1.4     2.0
  P1         17     19   PRD        FIT COVERS M/C 1                1.0   8MAY67  15MAY67   19MAY67   26MAY67      1.4     1.0
  P1         11     15              DUMMY                            .0  24APR67  24APR67   12MAY67   12MAY67      2.4      .0
  P1         15     18   PRD        SUB-ASSEMBLY CIRCUITRY          1.0  24APR67   1MAY67   12MAY67   19MAY67      2.4     2.0
  P1          2      4              LEAD                            1.0   9JAN67  16JAN67    3FEB67   10FEB67      3.4      .0
  P1          2     14   TEC        DESIGN CIRCUITS                 3.0   9JAN67  30JAN67    3FEB67   24FEB67      3.4      .0
  P1  L       4      5   PUR        OBTAIN RAW MAT'L                9.0  16JAN67  18MAR67   10FEB67   14APR67      3.4      .0
  P1          4     10              LEAD                            3.0  16JAN67   6FEB67   10FEB67    3MAR67      3.4     2.0
  P1  L       6      7   TEC        TOOL DESIGN                     6.0  16JAN67  27FEB67   20JAN67   22MAR67      3.4      .0
  P1         14     15 A PUR        OBTAIN B.O.ITEMS               11.0  30JAN67  17APR67   24FEB67   12MAY67      3.4      .0
  P1          7      9              LAG                             3.0  27FEB67  18MAR67   22MAR67   14APR67      3.4     1.0
  P1          5     11              LAG                             2.0  18MAR67   3APR67   14APR67   28APR67      3.4     3.0
  P1         13     12   PRD        MAKE BASE FRAMES                4.0  18MAR67  17APR67   14APR67   12MAY67      3.4     3.0
  P1         18     22   PRD        ST. 1 ASSY                      1.0  15MAY67  22MAY67   13JUN67   20JUN67      3.4     3.0
  P1         22     23   PRD        STAGE 2 ASSEM                   1.0  22MAY67  30MAY67   20JUN67   27JUN67      4.0      .0
  P3          2      5   TEC        PLAN COMPONENTS                 3.0  22DEC66  16JAN67   27JAN67   17FEB67      4.0     4.0
  P3          5      7   PRD        MAKE COMPONENTS                 6.0  16JAN67  27FEB67   17FEB67   31MAR67      4.4      .0
  P3  L       7      8   PRD        SUB-ASSY                        3.0  27FEB67  18MAR67   31MAR67   21APR67      4.4      .0
```

التقديرات الفنية :

بعد الانتهاء من الاتفاق على الشبكة يجب وضع التقديرات الخاصة بالزمن المقدر لكل نشاط وهذه التقديرات عادة ماتوضع على اساس الخبرة ، وفى بعض المشاريع ذات الطبيعة الخاصة مثل مشاريع الابحاث العلمية حيث لايمكن الجزم بالفترة الزمنية اللازمة للقيام ببعض الانشطة فانة يمكن وضع ثلاثة تقديرات :

أ ــ التقدير المتفائل ب ــ التقدير المتشائم جـ ــ التقدير المحتمل

وفى هذه الحالة يعطى هذه التقديرات الثلاث للحاسب بدلا من التقدير الواحد ويقوم الحاسب باخذ متوسطها باستخدام المعادلة الزمن = $\frac{\text{أ} + ٤\text{جـ} + \text{ب}}{٦}$ وهذا يجرى تقدير الزمن بوحدات نسبية مثل الاسبوع ، الوردية ، الساعة ، او الشهر .. الخ .

وهكذا فانه بعد بناء الشبكة ووضع التقديرات الزمنية يلزم تغذية المعلومات الى الحاسب وفى الايام الخالية كانت هذه العملية تتم بواسطة الكروت المثقبة اما الان فانها تتم بطريقة تخاطبية باستخدام مفاتيح الشاشات المرئية ومن الطبيعى ان الحاسب يستخدم البرنامج الموضوع له لدراسة شبكة البيرت لحساب النتائج واخراجها اما مطبوعة على شكل جداول او مرسومة بواسطة جهاز رسم خاص.

ولنتوقف هنا عند كلمة برنامج فلها اهمية خاصة وعلى كل فلنتركها فى الوقت الحالى فليست هى موضوعنا الاساسى ولنا عودة اليها ثانيةان شاء الله ، نعود الى النتائج يقوم الحاسب باختبار كل المسارات الممكنة خلال الشبكة ويحدد المسار الحرج ثم يقوم ببيان البرنامج الزمنى والسماح بالنسبة لكل من الانشطة والحوادث والسماح هو قيمة الزمن الذى يسمح لنشاط معين ان يزيد به على الفترة المقدرة له دون ان يؤدى ذلك الى الاضرار بتاريخ التسليم بالنسبة للمشروع ، وبالطبع فان النتائج المستخرجة من الحاسب تهيئ لمخطط المشروع الالمام بالمعلومات التى تفيده فى مراقبة سير التنفيذ حسب البرنامج واهم هذه النتائج

مايلـــــــــى :ـ

ـ المسار الحرج وهى قائمة بالنشاطات فى الشبكة التى يجب فيها البدء والانتهاء طبقا للمخطط الزمنى دون زيادة او نقصان اذا اريد اتمام المشروع فى وقته المحدد .

ـ جدول بكل النشاطات والسماح فى اجزاء الشبكة المختلفة ،موضحا به تواريخ اول وآخر بدء للنشاط Earliest Start & Latest Start وكذلك تواريخ اول وآخر انهاء للنشاط
Earliest Finish and Latest Finish of each activity

ويوضح شكل (١٤) احد التطبيقات الهندسية وفيه يستخدم شبكة بيرت لتخطيط ومراقبة انتاج احد اجزاء الالات كما يوضح شكل (١٢) صورة من جدول النتائج المستخرجة من الحاسب بالنسبة لمثال آخـــر .

هذا الجدول يمكن تمثيلة بصورة بيانية تمثل المسارات الثمانية عشرة الموجودة فى هذا المشروع ومنها يمكن استنتاج مايسمى بمنحنى التحميل الخاص بالوحدات الثلاث الرئيسية للمشروع
Work Load Histogram

ويتم تكوين منحنى التحميل باستخدام اول ابتداء لكل النشاطات او آخر ابتداء لكل النشاطات او نقطة تعتمد على الامكانيات فيما بين الطريقتين ولنوضح ذلك بتفصيل اكثـــر .

* استخراج منحنى التحميل باستخدام اول ابتداء - Aggregation of Work Load Histogram by Earlist Start

وفى هذه الحالة المبينة فى شكل (١٥) فان كل النشاطات يجرى بدأها فى اقرب وقت ممكن لبدئها دون النظر للامكانيات المتاحة فى المشروع وهذا يعطى فكرة نظرية عن مستوى المطلوب من الامكانيات للمشروع فى هذه الحالة ، مثل هذا المخطط يحمل طابعا تضخميا بالنسبة للتكاليـــف حيث ان النشاطات يتم انجازها قبل الحاجة الفعلية لها دون مبرر ،غير ان هذا يضمن انجاز المشروع فى الوقت المحدد له مع تقليل المخاطرة بانهاء المشروع بعد موعده ،وتوضح المنحنيات المبينة فى هذا الشكل تحميلا غير منتظم وغير معقول غير ان الدراسة النظرية لهذا النوع من

المنحنيات يفيد ادارة المشروع فى تقييم الامكانيات المطلوبة٠

* استخراج منحنى التحميل باستخدام آخر ابتداء
Aggregation of Work Load Histogram by Latest Start

وفى هذه الحالة المبينة فى شكل (١٦) فان كل النشاطات يجرى بداها فى آخر وقت ممكن وفى هذه الحالة فان تكلفة المشروع تكون اقل مايمكن غير ان المخاطرة كبيرة فى اى نشاط ان يتأخر عن الجدول الزمنى وغالبا مايتأخر تنفيذ المشروع ومايترتب علية من غرامات (ملحوظة) :

قد يلجأ بعض الناس الى هذا الحل اذا كانت قيمة غرامة التأخير اقل من الوفر فى النفقات) ٠

وفى هذه الحالة ايضا فان منحنى التحميل الناتج يكون ايضا غير منتظم ولا يمكن قبولة من ناحية الامكانيات٠

* استخراج منحنى التحميل مع أخذ الامكانيات فى الاعتبار
Aggregation of Work Load Histogram by Resource Allocation

فيما بين النهايتين المذكورتين سابقا يمكن العثور على منحنى تحميل يكون اكثر انتظاما ويأخذ الامكانيات المتاحة فى الاعتبار ـ استخدام عدد شبة ثابت من العمال ، عدم اللجوء لورديات اضافية ٠٠ الخ ٠ ـ ويبين شكل (١٧) كيف اعيد تخطيط النشاطات لتحقيق عدم زيادة التحميل فى اى من الاجزاء الثلاثة الرئيسية للمشروع عن الطاقة المتاحة فى كل منها وطبقا لهذا فقد تأخر تنفيذ المشروع اسبوعا عن المحدد له ولكن رغم ذلك التأخير فان عنبر الماكينات لم يحدث ابداً ان حمل طاقته او حتى حمل بكل طاقته وهذا وضع طبيعى ذلك ان المعدل الذى يجرى به تصميم الالات هو الذى يحدد عدد العمال فى عنبر الالات فى اى وقت من الاوقات٠

الى هنا ننتهى من طريقة المسار الحرج فى دراسة المشروعات ونعود الى برنامج الحاسب

عودة الى البرنامج والبرمجة :

لن اتوه معكم فى المصطلحات من امثال الجول (Algarthnisc Langauage (Algol 60

فورتران (Formula Translator Fortran 4 ، كوبول (Cabel)
Commercial Business Oriented Language
Symbolic
Beginners All-purpose Symbolic (Basic)

او بيزك (

الى اخر هذا الطوفان من التعبيرات واللغات والمصطلحات المهمان برامج الحاسبات موجودة ومعلبة بصورة تخاطبية بحيث يتخاطب الحاسب مع المستخدم وايا كانت اللغة المكتوب بها البرنامج فان البرنامج يجب ان يوفر فيه الوضوح بحيث يستطيع الفرد العادى ان يستخدمة بسهولة ، ولقد سمحت الفترة الطويلة منذ اختراع الحاسبات الضخمة ذات الجيوش الجرارة من اطقم الخدمة والبرمجة الى العصر الحالى — عصر الحاسبات الالكترونية الشخصية — سمحت هذه الفترة بوضع برامج لكل التطبيقات التى تخطر على بال المستخدم العادى وهى تباع معلبة على اقراص باثمان زهيدة لمستخدمى الحاسبات الشخصية كما تباع باثمان رهيبة لمستخدمى الحاسبات الكبيرة وعلى كل مالنا ولهذا ، ومع ذلك فليست البرمجة فى حد ذاتها طلسما من الطلاسم ، كما ان لغات المحاسب ليست سحرا من التنزيل ان هى الا بضع ساعات يقضيها من اراد تعلمها ويستطيع ان يكتب بنفسة البرامج التى يشاء لحل المشاكل التى تقابلة ومع ذلك فاننى على عكس الكثيرين لا انصح بتعلم البرمجة الا على سبيل الهواية .. او اذا لم توجد للتطبيق المراد دراسته برامج معلبة — وهذه حالة قد اصبحت نادرة جدا — ولا يلجأ اليها الا المتخصصين والان دعنا نجرب كتابة برنامج صغير على سبيل الهواية بلغة بيزك .

<u>لغـــــة بيزك</u> : تبلغ عدد الكلمات هذه اللغة حوالى عشرون كلمة اهمها

Fom...Nex, Let, go to, if go Sub (Routine), Oim(ension), Print, Road, Input open,)file), Step,...etc

يمكن اذا استخدمها الانسان بعناية ان يتعلم كتابة برنامج فى ظرف ٨ ثمانية ساعات ، وهى اللغة مكونة من جمل كل منها يبدأ برقم تسمى كل منها Statement ويجرى تنفيذ البرنامج حسب تسلسل هذه الارقـــــام .

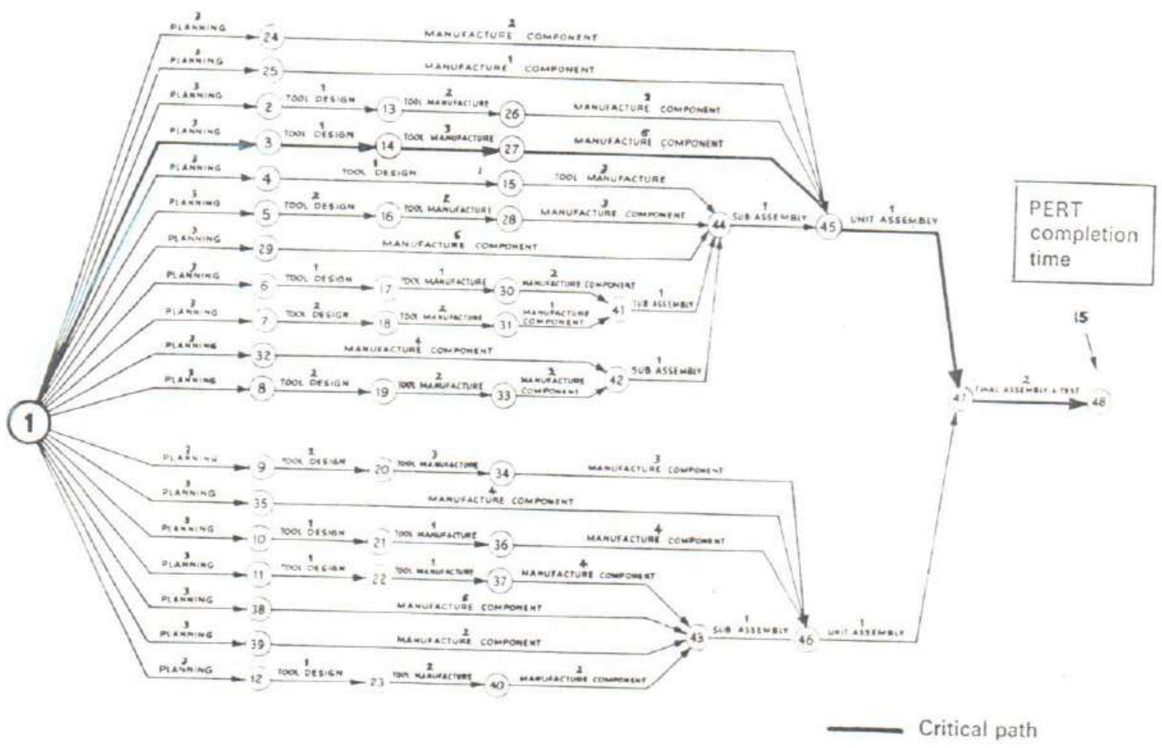
برنامج لحساب المتوسط لعدد خمسة ارقام وتكرار ذلك الف مرة

```
10 For   I= 1 to 1000
15 Let   S= 0
20, For  J= 1 to 5

30 Read  A(J)
40 Let   S=S+A(J)
45 Next  J

50 Let   M=S/5
55 Primt I,M
60 Next  I
70 End
```

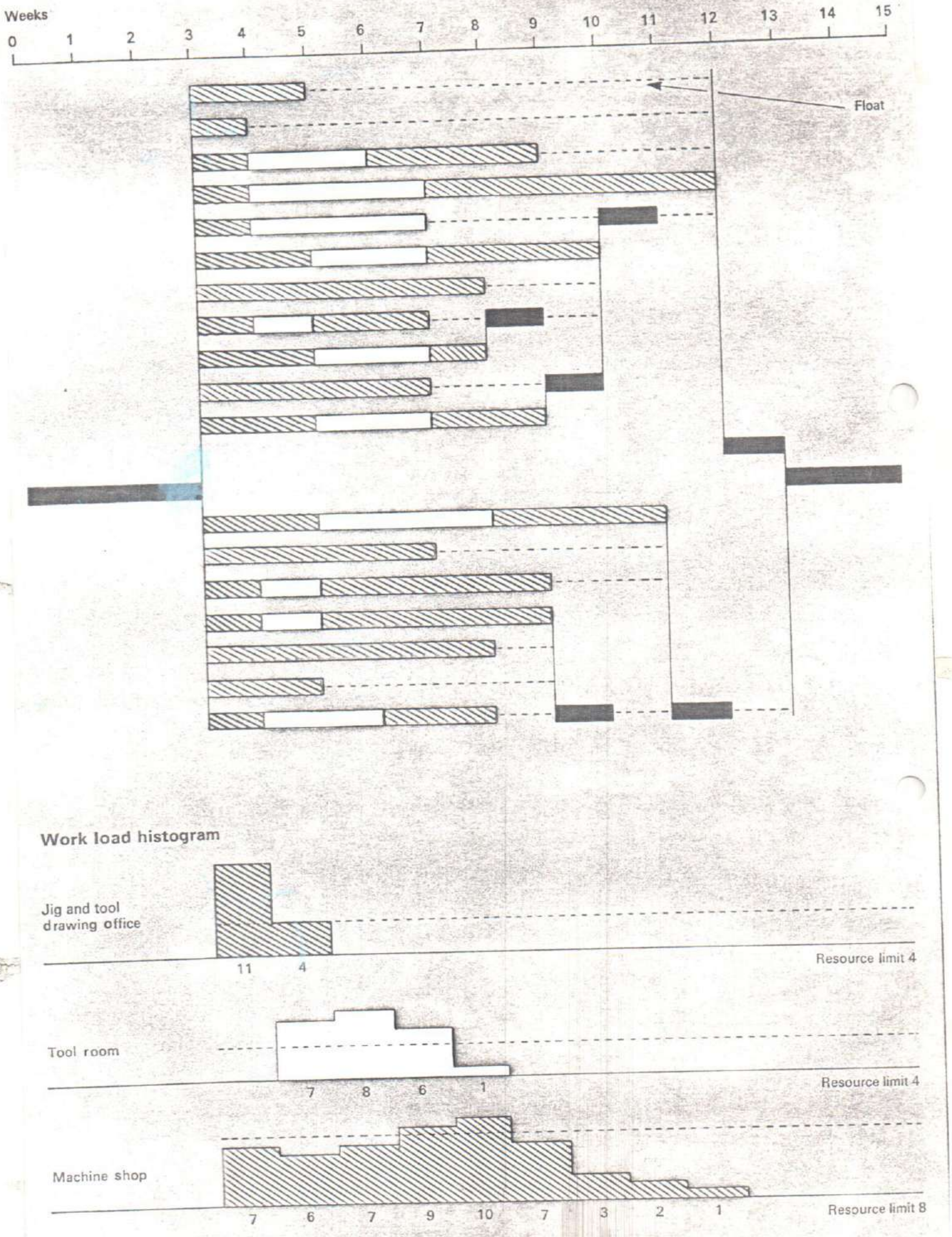
هذا البرنامج لايحتاج فى شرحة الى اى تعليق.

تذييل :

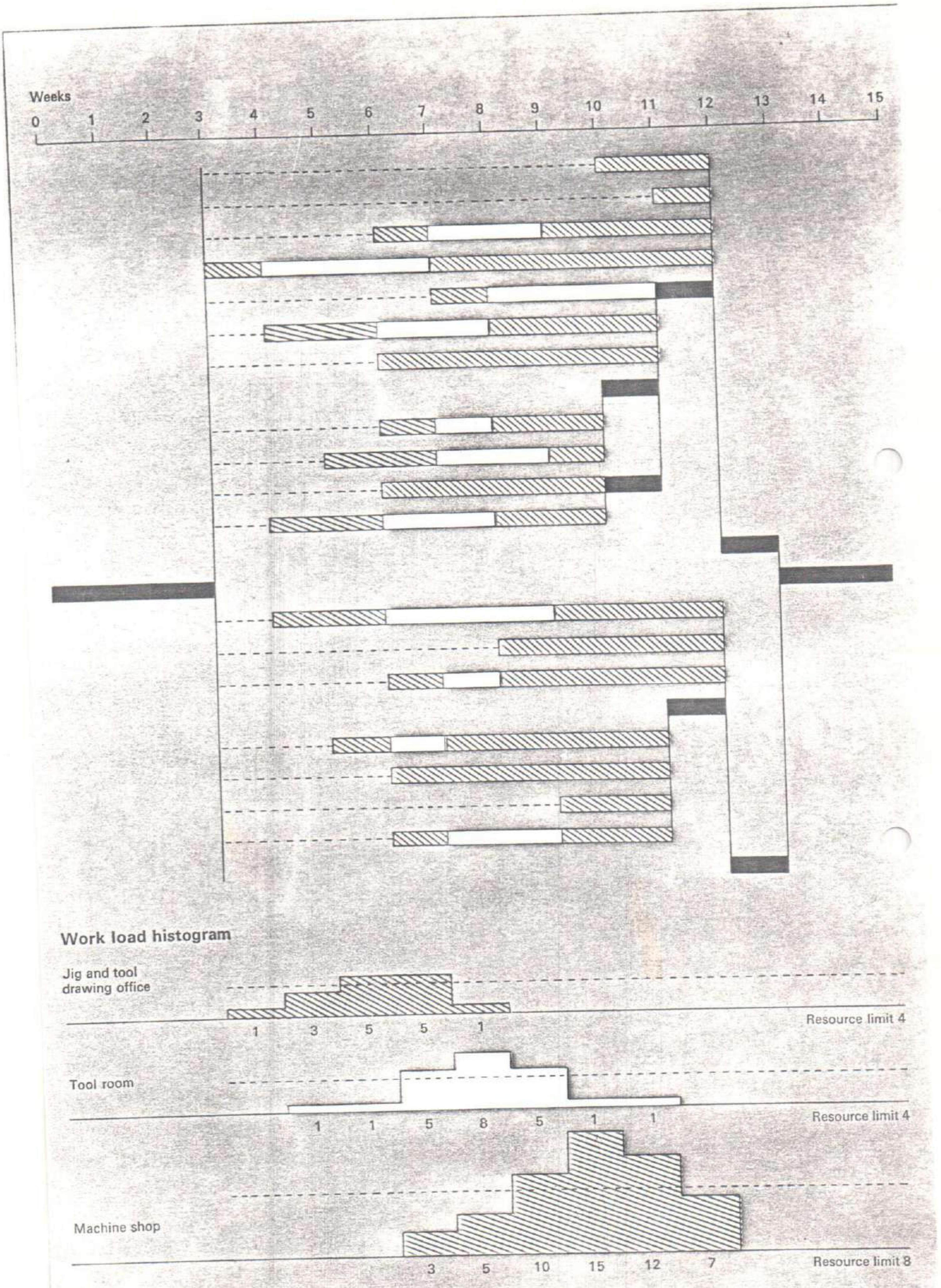
والان وقد احطنا بفكرة لابأس بها عن الحاسبات الشخصية واستخدامها فى دراسة المسار الحرج وبعد ان نرى التطبيق العلمى على الجهاز الحاسب الشخصى ، وبعد ان اخذنا فكرة عن اسماء لغات الحاسب واخذنا مثالا مكتوبا بلغة بيزك اسمحوا لى كتذييل للموضع ان اورد برنامجا كاملا معلبا يبلغ ثمن شراوءه حوالى عشرة دولارات اذا اشترى مع حاسب شخصى بينما كانت تبيعه شركة .I.C.L بثلاثة الاف دولار مكتوبا بلغة اخرى لحساب المسار الحرج لاى مشروع من المشاريع وهو مكتوب ايضا بلغة بيزك ومرفق به حل لاحد الامثلة فى بناء البيوت.



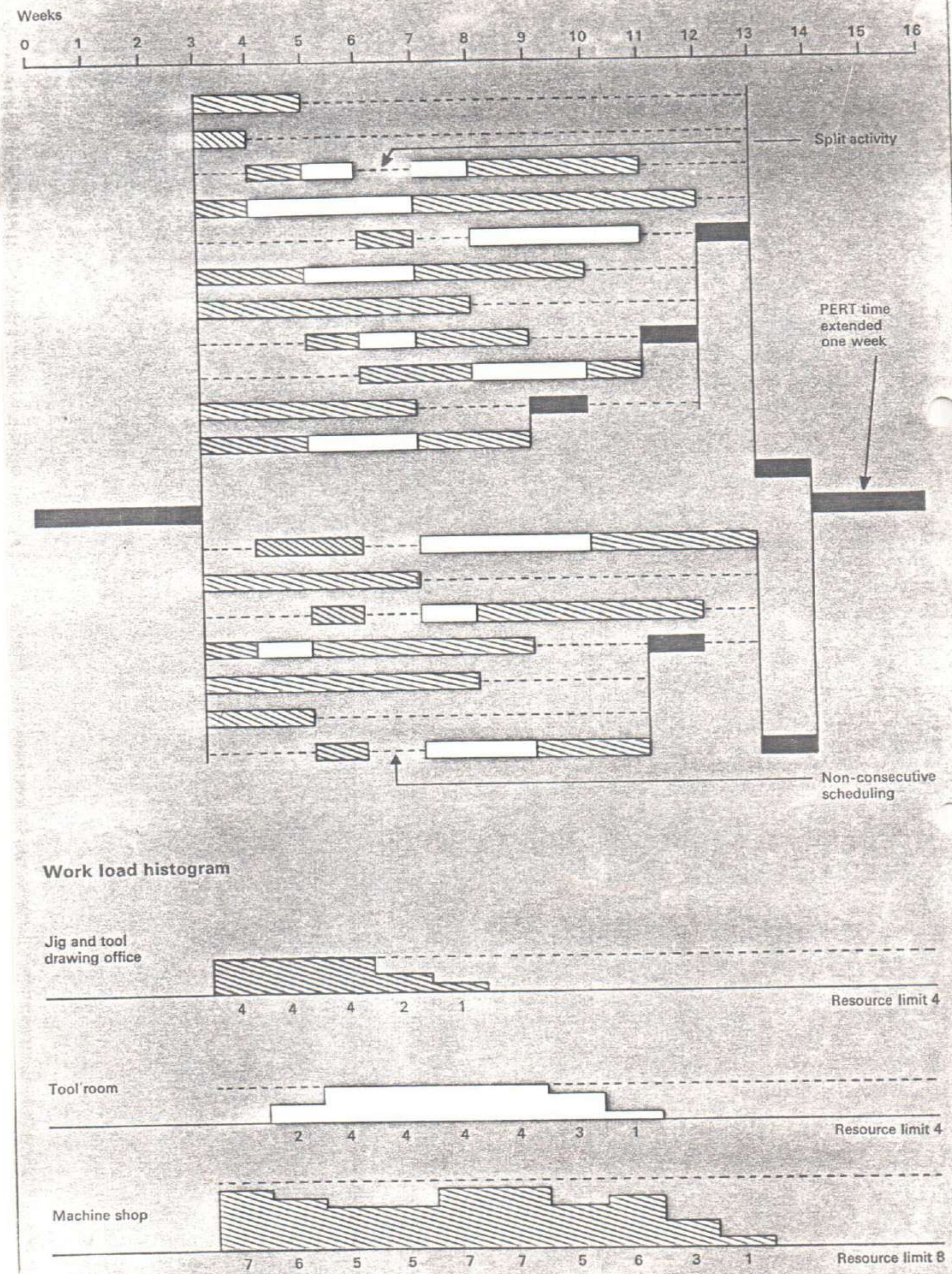
PERT completion time
15
Critical path



15 Earliest start



16 Latest start



17 Resource allocation

```
00001 REM THIS PROGRAM WILL ILLUSTRATE THE SIMPLE EVENT-ORIENTED
00002 REM CRITIAL PATH CALCULATIONS.  IT IS ASSUMED THAT THE NETWORK
00003 REM IS CORRECT SINCE NO VERIFICATION IS DONE.  THIS PROGRAM WORKS
00004 REM CURRENTLY WHEN IT HAS ONE START AND ONE END.  THE SEQUENCE OF
00005 REM OF THE ACTIVITIES CAN AND WILL CHANGE THE RESULTS SOMETIMES.
00006 REM IT IS BEST TO HAVE ACTIVITIES IN SEQUENCE BY START EVENT AND
00007 REM FINISH EVENT.
00008 REM >>THE PROGRAMS PRODUCES RELATIVE TIME UNITS<<
00010 DIM I(100),J(100),A(100),B(100),T(100),C(100)
00020 DIM E(100),F(100),G(100),H(100),M(100),P(100)
00028 REM READ START AND FINISH DATE, NUMBER OF EVENTS
00030 READ S,D,N
00040 FOR K= 1 TO N
00048 REM START NODE,END NODE, DUR OF ACTIVITITY
00050  READ I(K),J(K),T(K)
00060 NEXT K
00070 FOR K= 1 TO 100
00080  LET H(K)=S
00090  LET P(K)=D
00100 NEXT K
00110 LET M1=0
00120 FOR K= 1 TO N
00130  LET N1=I(K)
00140  LET N2=J(K)
00150  LET A(K)=H(N1)
00160  LET C(K)=A(K)+T(K)
00170  LET Z1=C(K)-H(N2)
00180   IF Z1<=0 THEN 200
00190  LET H(N2)=C(K)
00200  LET Z1=C(K)-M1
00210   IF Z1<=0 THEN 230
00220  M1=C(K)
00230 NEXT K
00240 FOR K=1 TO N
00250  L=N+1-K
00260  LET N1=I(L)
00270  LET N2=J(L)
00280  LET E(L)=P(N2)
00290  LET B(L)=E(L)-T(L)
00300  LET Z1=B(L)-P(N1)
00310   IF Z1>=0 THEN 330
00320  LET P(N1)=B(L)
00330  LET F(L)=B(L)-A(L)
00340 NEXT K
00350 FOR K=1 TO N
00360  LET N2=J(K)
00370  LET G(K)=F(K)-P(N2)+H(N2)
00375 NEXT K
00377 PRINT ´HOUSE CONSTRUCTION PROJECT(IN DAYS)--EXPEDITED TIME´
00378 PRINT
00379 PRINT ´ START   FIN   DLR    ES    LS    EF    LF    TS    FS´
00380 FOR K= 1 TO N
00381  LET K9=K9+1
00385 READ X$
00386 PRINT X$
00390 PRINT IN FORM´%%%%%%´:I(K),J(K),T(K),A(K),B(K),C(K),E(K),F(K),G(K)
00393 IF K9 <=25 THEN 400
00394  LET K9=0
00395 PRINT ´HOUSE CONSTRUCTION PROJECT(IN DAYS)--EXPEDITED TIME´
00396 PRINT
```

```
00397 PRINT ' START   FIN   DLR   ES   LS   EF   LF   TS   FS'
00400 NEXT K
00500 DATA 1,26,31
00501 DATA 0,1,4
00502 DATA 1,2,2
00503 DATA 2,3,4
00504 DATA 3,4,6
00505 DATA 2,5,1
00506 DATA 5,6,2
00507 DATA 5,7,3
00508 DATA 3,8,2
00509 DATA 3,9,4
00510 DATA 6,9,0
00511 DATA 7,10,1
00512 DATA 8,10,0
00513 DATA 9,10,0
00514 DATA 10,11,3
00515 DATA 11,12,1
00516 DATA 11,13,2
00517 DATA 11,14,3
00518 DATA 4,15,2
00519 DATA 15,16,1
00520 DATA 2,17,1
00521 DATA 14,18,2
00522 DATA 12,19,3
00523 DATA 13,19,0
00524 DATA 19,18,0
00525 DATA 19,20,1
00526 DATA 16,21,2
00527 DATA 17,21,0
00528 DATA 21,22,5
00529 DATA 22,23,0
00530 DATA 18,23,0
00531 DATA 20,23,0
00600 REM DISC IN SAME SEQUENCE AS ABOVE DATE NETWORK
00601 DATA EXCAVATE & POUR FOOTERS
00602 DATA POUR CONCRETE FOUNDATIONS
00603 DATA ERECT FRAME & ROOF
00604 DATA LAY BRICKWORK
00605 DATA INSTALL DRAINS
00606 DATA POUR BASEMENT FLOOR
00607 DATA INSTALL ROUGH PLUMEING
00608 DATA INSTALL ROUGH WIRING
00609 DATA INSTALL A/C
00610 DATA A/C WAIT FOR BASMENT FLOOR
00611 DATA FASTEN PLASTER & PLASTER BD
00612 DATA PLASTER WAIT FOR WIRING
00613 DATA PLASTER WAIT FOR A/C
00614 DATA LAY FINISH FLOORING
00615 DATA INSTALL KITCHEN EQUIPMENT
00616 DATA INSTALL FINISHED PLUMBING
00617 DATA FINISH CARPENTY
00618 DATA FINISH ROOFING & FLASHING
00619 DATA FASTEN GUTTERS & DOWNSPOUTS
00620 DATA LAY STORM DRAINS
00621 DATA SAND & VARISH FLOORS
00622 DATA PAINTING
00623 DATA PAINTING WAIT FOR PLUMBING
00624 DATA FIN FLOOR WAIT FOR PAINTING
00625 DATA FINISH ELECTRICAL WORK
```

```
00626 DATA FINISH GRADING
00627 DATA GRADING WAIT FOR STRM DRAINS
00628 DATA POUR WALKS & LANDSCAPE
00629 DATA COMPLETE OUTSIDE WORK
00630 DATA COMPLETE INSIDE WORK
00631 DATA COMPLETE INSIDE WORK
00800 REM  ES--EARLY START
00801 REM  LS--LATEST START
00802 REM  EF--EARLY FINISH
00803 REM  LF--LATEST FINISH
00804 REM  TS--TOTAL SLACK
00805 REM  FS--FREE SLACK
00999 END
```

```
INSTALL ROUGH WIRING
    3     8     2    10    16    12    18     6     0
INSTALL A/C
    3     9     4    10    14    14    18     4     0
A/C WAIT FOR BASMENT FLOOR
    6     9     0     9    18     9    18     9     5
FASTEN PLASTER & PLASTER BD
    7    10     1    10    17    11    18     7     3
PLASTER WAIT FOR WIRING
    8    10     0    12    18    12    18     6     2
PLASTER WAIT FOR A/C
    9    10     0    14    18    14    18     4     0
LAY FINISH FLOORING
   10    11     3    14    18    17    21     4     0
INSTALL KITCHEN EGUIPMENT
   11    12     1    17    21    18    22     4     0
INSTALL FINISHED PLUMBING
   11    13     2    17    23    19    25     6     0
FINISH CARPENTY
   11    14     3    17    21    20    24     4     0
FINISH ROOFING & FLASHING
    4    15     2    16    16    18    18     0     0
FASTEN GUTTERS & DOWNSPOUTS
   15    16     1    18    18    19    19     0     0
LAY STORM DRAINS
    2    17     1     6    20     7    21    14     0
SAND & VARISH FLOORS
   14    18     2    20    24    22    26     4     0
PAINTING
   12    19     3    18    22    21    25     4     0
PAINTING WAIT FOR PLUMBING
   13    19     0    19    25    19    25     6     2
FIN FLOOR WAIT FOR PAINTING
   19    18     0    21    26    21    26     5     1
FINISH ELECTRICAL WORK
   19    20     1    21    25    22    26     4     0
FINISH GRADING
   16    21     2    19    19    21    21     0     0
```

Peter R. Ohs Page 5 of 7
(Example 2)

```
HOUSE CONSTRUCTION PROJECT(IN DAYS)--EXPEDITED TIME

START   FIN   DUR    ES    LS    EF    LF    TS    FS
GRADING WAIT FOR STRM DRAINS
   17    21     0     7    21     7    21    14    14
POUR WALKS & LANDSCAPE
   21    22     5    21    21    26    26     0     0
COMPLETE OUTSIDE WORK
   22    23     0    26    26    26    26     0     0
COMPLETE INSIDE WORK
   18    23     0    22    26    22    26     4     4
COMPLETE INSIDE WORK
   20    23     0    22    26    22    26     4     4

TIME :   .413
```

```
HOUSE CONSTRUCTION PROJECT(IN DAYS)--EXPEDITED TIME

  START   FIN    DUR     ES      LS      EF      LF      TS      FS
EXCAVATE & POUR FOOTERS
      0     1      4      0       0       4       4       0       0
POUR CONCRETE FOUNDATIONS
      1     2      2      4       4       6       6       0       0
ERECT FRAME & ROOF
      2     3      4      6       6      10      10       0       0
LAY BRICKWORK
      3     4      6     10      10      16      16       0       0
INSTALL DRAINS
      2     5      1      6      13       7      14       7       0
POUR BASEMENT FLOOR
      5     6      2      7      16       9      18       9       0
INSTALL ROUGH PLUMBING
      5     7      3      7      14      10      17       7       0
```

ملحق

اسعار الجهاز فوب مطار القاهرة

	الثمن بالدولار
أولا الوحدة المركزيـــة	
أ ـ الوحدة المركزية بالكابينة ومصدر التيار من طراز آى سى اى ٨٠	٣٦٠٠
ب ـ الوحدة المركزية بالكابينة ومصدر التيار من طراز آى سى اى ٨٠٠٠	٥٢٠٠
ثانيا : الذاكرة الداخلية :	
أ ـ كارت سعة ٣٢ ألف حرف	٥٠٠
ب ـ كارت سعة ٦٤ ألف حرف	١٠٠٠
جـ ـ كارت سعة ٢٥٦ ألف حرف	٣٢٠٠
ثالثا : الذاكرة الثانوية	
أ ـ جهاز اقراص مرنة $\frac{١}{٤}$ ٥ بوصة سعة ٣٠٠ الف حرف كامل بلوح التحكم الخاص بأربعة اجهزة اقراص (كل جهاز اضافى ٦٠٠ دولار)	٩٠٠
ب ـ جهاز اقراص مرنة ٨ بوصة سعة ٥٠٠ ألف حرف كامل بلوح التحكم الخاص بأربعة اجهزة كل جهاز اضافى ٨٠٠ دولار)	١٢٠٠
جـ ـ جهاز اقراص ثابتة سعة ١٠ مليون حرف كامل بلوح التحكم لاربعة اجهزة (كل جهاز اضافى ٤٠٠٠)	٥٠٠٠

د — جهاز اقراص ثابتة سعة ٢٠ مليون حرف كامل بلوح التحكم لاربعة اجهزة (كل جهاز اضافى ٢٥٠٠ دولار)	٦٥٠٠
هـ — جهاز شرائط مغناطيسية سعة ١٥ مليون حرف بلوح التحكم الخاص به	٤٧٠٠
ز — جهاز اقراص ثابتة سعة ٣٠ مليون حرف كامل بلوح التحكم لاربعة اجهزة (كل جهاز اضافى ٦٠٠٠ دولار)	٧٠٠٠
رابعا : شاشة تليفزيونية سعة ٢٤ سطرا كل سطر ٨٠ حرفا للكتابة باللغتين العربية والانجليزية كاملة بلوح التحكم ولوحة مفاتيح الادخال والاخراج	٢٠٠٠
خامسا : أ — طابع اسطر على ورق فولسكاب سرعة ٨٠ حرف/ ثانية	١٠٠٠
ب — طابع اسطر على ورق عريض (١٥ بوصة) سرعة ١٥٠ حرف/ ثانية كامل بلوح التحكم	٣٠٠٠
ج — طابع اسطر على ورق عريض (١٥ بوصة) سرعة ٤٥٠ حرف / ثانية كامل بلوح التحكم	٧٠٠٠
د — جهاز رسم منحنيات واشكال هندسية بالحبر الشينى	٢٥٠٠
سادسا : أ — نظام وبرامج التشغيل ومترجم لغة بيزك ولغة فورتران للطراز آى سى اى — X٠	١٥٠٠
ج — مجموعة (حزمة) برامج تطبيقات هندسية وتجارية (تشمل بيرت — سى بى ام — وبرامج هندسية انشائية)	١٥٠٠

أ — التركيب وتدريب العاملين مجانا .
ب — الصيانة مجانا خلال فترة الضمان .

```
TO C.G. OF FIRST ROW AND C.G. OF CABLE GROUP ARE
 ? 100,226.6
NUMBER IN ROW  1  ? 1
NUMBER IN ROW  2  ? 1
NUMBER IN ROW  3  ? 1

DISTANCE TO CENTRE OF FIRST ROW IS 100 MM

DISTANCE BETWEEN CENTRES OF ROWS IS 126.6 MM

MINIMUM END BLOCK WIDTH TO ACCOMMODATE
ANCHORAGES= 280 MM

IF ALL ARRANGEMENTS HAVE BEEN CONSIDERED
THEN TYPE 1 ELSE 0 ? 1



RUNNING TIME:     1.2 SECS   I/O TIME :    3.1 SECS


PBS

REQUIRED MULT (KNM)= ? 691.62
YOUR VALUES OF OVERALL BEAM DEPTH,
UPPER FLANGE WIDTH,
OUTER AND INNER FLANGE THICKNESSES
AND WEB THICKNESS (MM) ARE
 ? 900,500,200,225,150
CHARACTERISTIC CONCRETE STRENGTH (N/MM↑2)
AND LOSS FACTOR ARE ? 50,0.8
NUMBER OF CABLES= ? 3
CABLE TYPE - NUMBER OF STRANDS/CABLE
AND STRAND DIAMETER (MM) ARE ? 4,12.5
HOW MANY ROWS ? 3
NUMBER OF CABLES IN ROW 1  ? 1
NUMBER OF CABLES IN ROW 2  ? 1
NUMBER OF CABLES IN ROW 3  ? 1
DISTANCE (MM) FROM BOTTOM OF BEAM
TO C.G. OF FIRST ROW= ? 100
DISTANCE (MM) BETWEEN ROWS= ? 125

ACTUAL MULT GREATER THAN REQUIRED

ULTIMATE MOMENT OF RESISTANCE= 1009.76 KNM

DEPTH OF NEUTRAL AXIS= 171.195 MM

STEEL STRESS AT ULTIMATE LOAD EXPRESSED AS A
PERCENTAGE OF THE DESIGN STRENGTH

AT ROW 1        99 %
AT ROW 2        99 %
AT ROW 3        98 %


RUNNING TIME:     2.4 SECS   I/O TIME :    4.3 SECS
```

Chapter 4

The Computer Aided Design of Rigid Frames—Aspects Common to All Frames

4.1 Introduction

The design of a member from an indeterminate structure is not significantly different from that of its counterpart in a determinate structure; the design requirements for both are essentially the same, but in the former case they are more difficult to assess because the very proportions of each member affect in some degree the behaviour of every other member in the structure. In a structure comprising a large number of members the practical effects of changing the section proportions of a single member are fortunately local—but the effects are often significant within the sphere of influence of the modified member. This interdependence of parts poses a problem common to both the manual and computer aided design of continuous frames alike—that of initially assessing the relative second moments of area of the members comprising the structure. If precisely correct values can be assigned initially then the structure is effectively designed before a formal solution is even begun. But this is a counsel of perfection. In practice rigid frame design becomes an iterative process; one from which an acceptable solution will always be forthcoming as the outcome of a cycle of design iterations in which each iteration produces an increasingly better approximation.

The manner in which starting values for second moments of area are assessed is influenced by the type of design calculation in which they are to be used. For an automatic iterative calculation it is almost sufficient to say that any initial approximation is better than none at all. Indeed, it will be shown in Chapter 6 that in the case of designs involving the choice of steel sections from a standard list an automatic solution will always be possible and the initial values assigned to the second moments of area are largely immaterial to the quality of the final result. If for reinforced concrete structures a different set of constraints is accepted from those implied by a list of standard sections, e.g. designer specified rib and column proportions and steel ratios, then these too will furnish sufficient data to lead to a successful automatic design conclusion. In this case the starting values for the second moments of area are conveniently based upon sections which are derived from maximum allowable deflection and slenderness criteria. In comparison with steel structures, however, it must be expected that concrete structures will require a larger number of design iterations to attain a solution because of the significant contribution that the (initially unknown) structural self-weight makes towards the total loading.

In contrast with this, decision design based programs require informed initial guesses to be made in order to reasonably restrict the overall design time. In the case of continuous reinforced concrete frames this is not such a confining requirement when it is appreciated that the concrete dimensions (which to a degree affect the distribution of force actions) themselves represent a wide range of strengths (resistance to bending, shear, axial load and torsion) depending upon the type and quantity of reinforcement incorporated. During the design of such structures it often happens that a set of guessed section dimensions does not, on analysis of the structure, produce the force actions for which the sections were initially proportioned; but because of their inherent strength range they may still be reinforced to sustain the new force actions. In both automatic design and decision design it is the finite step in strength between one practical section and the next that allows convergence to a unique solution.

It must be accepted that structural analysis is an important aspect of the overall design process and that it is more necessary to choose a method which reflects the behaviour of the structural material than it is to make a choice which is based purely on its suitability for computer programming and operation. Whilst stiffness matrix analysis lends itself to compact programming procedures and efficient computer solution most available matrix analysis programs are based upon assumed linear behaviour. Their results are only directly applicable to a material such as steel working within its elastic range. In the case of reinforced concrete, its non-linear stress/strain characteristics, its creep, shrinkage, cracked regions and non-uniform distribution of reinforcement all conspire to produce deformations which are at times grossly at variance with those predicted by linear elastic theory. When dealing with such materials it might be wiser either to favour a less sophisticated analytical method (albeit one which is probably based upon assumed elastic behaviour) and to temper the more obvious discrepancies between theoretical and actual behaviour with engineering judgement, or to opt for even greater sophistication than that offered by linear matrix analysis by taking account of non-linear behaviour.

Anyone able to specify the parameters of a design may use an automatic design based program successfully, but only competent designers will be capable of producing reasonable results from a decision design based program. This follows from the fact that in decision design the computer is essentially nothing more than an aid to calculation; it extends the designer's powers but abrogates few of his responsibilities.

4.2 Describing the Structure and its Loading to the Computer

4.2.1 Regular Rectangular Frames

Multi-storey, multi-bay frames have a regular geometry and distribution of joints which make the assignment of structural information to the computer, and its subsequent handling and transfer, a relatively easy task. Each member, joint or force action—or indeed any quantity which may be used to describe the structure or its behaviour—can be represented by the content of an element in a rectangular array. Moreover, the array element and the quantity which

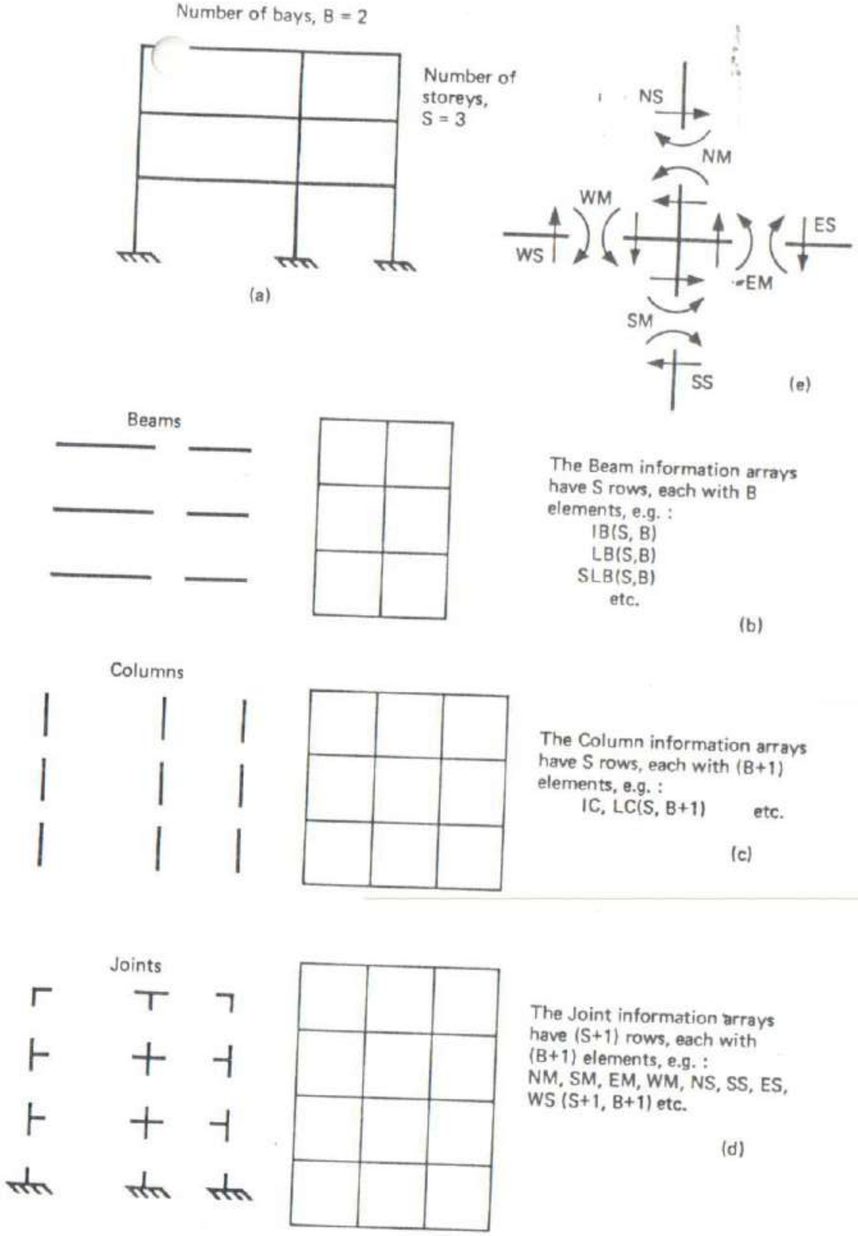


Figure 4.1 Arrays associated with the definition of multi-bay, multi-storey frames

it represents occupy the same relative positions in the array and the structure respectively. This one-to-one relationship between the structure and its associated arrays greatly simplifies the identification of numerical information with its origin in the structure.

Neither joints nor members need to be formally numbered. Their locations in the frame are known when the integer variables I and J are given values which specify the floor or storey level and the bay or vertical column line at which arithmetical operations are to be carried out.

If the number of storeys in a frame is denoted by S, and the number of bays by B, then the structure will contain S∗B beams. For the frame shown in Fig. 4.1a, S∗B = 6. Some information pertaining to these beams may be conveniently held in 6-element arrays comprising 3 rows and 2 columns. Thus SLB( ), IB( ) and LB( )—see Fig. 4.1b—could be arrays which hold information concerning the beam superimposed loads, second moments of area and spans, respectively. (It should be noted that in all but the simplest programming languages the variables and array names may take more complex descriptive titles than are allowed in BASIC, a facility which greatly simplifies the understanding of the printed program.)

If the integer variables I and J are each made equal to 2 say, to denote that we are interested in the array element defined by the intersection of the second row and the second column, then the content of LB(2, 2) will be the value of the span at the second floor level in the right hand bay. More frequently we would need to refer to the general element LB(I, J) and the other quantities associated with it such as IB(I, J) and SLB(I, J).

The frame also has (S + 1)∗(B + 1) joints and S∗(B + 1) columns. Associated joint arrays will therefore have (S + 1) rows and (B + 1) columns whilst the column member arrays will have S rows and (B + 1) columns—see Figs. 4.1c and d.

Using the convention that at a four-member joint the ends of the members are labelled North, South, East and West (see Fig. 4.1e), it is convenient to identify the bending moments at the ends of the members by NM, SM, EM and WM. In a similar way the shear forces which act at the ends of members are called NS, SS, ES and WS. Each family of force actions is stored in its own array which has (S + 1) rows and (B + 1) columns (see Fig. 4.1d). The force arrays will necessarily contain a number of permanent zero values. For example, all the elements in the first row of NM( ) will be zero because there are no frame columns above roof level.

In order to make calculations which involve the elements from two or more arrays it is necessary to establish relationships between the positions of elements in those arrays. This is most readily done by inspection. Referring to Figs. 4.1b, c and d it can be seen that the superimposed load on the beam identified by LB(I, J) is SLB(I, J). The bending moment at the left hand end of this member is EM(I, J) and at the right hand end, WM(I, J + 1). NM(I, J) is the bending moment at the lower end of the column member whose length is LC(I − 1, J).

By calculating the shear force at the end of a beam as a function of the beam span, the load upon it and the support bending moments the reader may verify the following relationship for $1 \leqq I \leq S$ and $1 < J \leqq (B+1)$:

$$\begin{aligned} WS(I, J) = {} & SLB(I, J-1) * LB(I, J-1)/2 - (EM(I, J-1) \\ & + WM(I, J))/LB(I, J-1) \end{aligned}$$

4.2.2 Irregular Frames

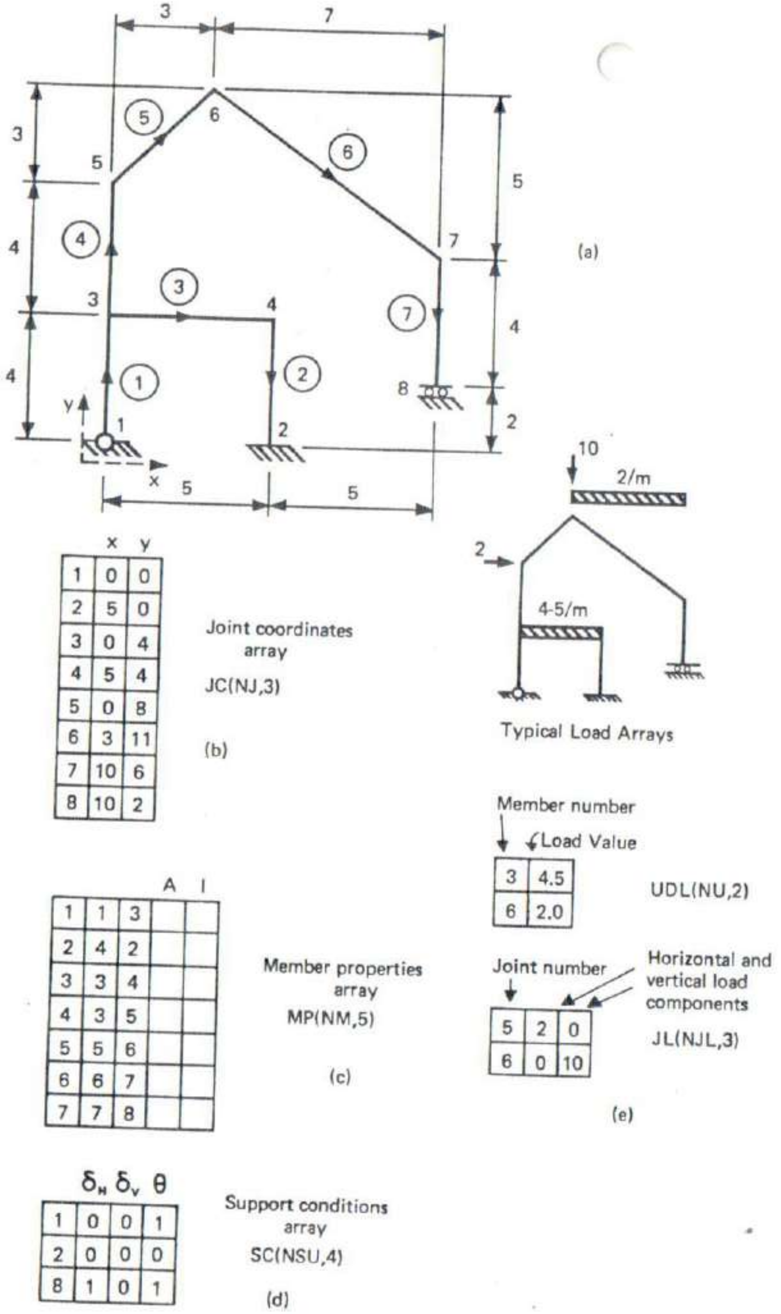
In contrast with the approach which was discussed in Section 4.2.1 a more general method of assigning structural information to the computer may be used. This method, which is suitable for both geometrically regular and irregular frames alike, is based upon a joint numbering system; its main application is in conjunction with matrix methods of analysis. Whilst in theory the order in which joints are numbered should not matter, in practice the numbering system has a significant effect on the size of the stiffness matrix array. An effort should therefore be made to number the joints in such a way that the maximum numerical difference between the two joints at the ends of any member is kept to a minimum.

An irregular frame is shown in Fig. 4.2a. It comprises 8 joints and 7 members; member number (3) for example is connected to joint numbers (3) and (4)—the direction in which the member is to be considered is indicated by an arrow.

Three arrays of data inform the computer of the frame's geometry, member properties and support conditions. These arrays, called JC( ), MP( ) and SC( ), are shown in Figs. 4.2b, c, and d. For a structure with NJ joints, array JC( ) has NJ rows and 3 columns. The first column records joint reference numbers in their correct order. The remaining two columns hold the associated *x* and *y* joint coordinates—in this case taken from joint number (1) as datum.

For a structure comprising NM members the second array, called MP( ), has NM rows and 5 columns. The member reference numbers are held in the first column of this array in their correct order. The remaining elements in a row contain the two joint reference numbers which define the ends of the member (entered in the order defined by the member direction), the cross-sectional area of the member and its second moment of area. For the initial analysis of an iterative design solution it is probable that A and I would be assigned programmed values. In this, two options are possible: either A and I are related to actual sections or the cross-sectional areas are given artificially inflated values which would have the effect of virtually suppressing the secondary force actions due to axial strains. This means that the initial second moments of area could be treated as relative rather than absolute values. Following the first round of analysis and section selection more realistic assessments of A and I would then be available for subsequent analyses.

With these two arrays the computer has information concerning the location, orientation and properties of all the members in the structure. One instance in which information from these arrays must be related occurs when the member



	x	y
1	0	0
2	5	0
3	0	4
4	5	4
5	0	8
6	3	11
7	10	6
8	10	2

			A	I
1	1	3		
2	4	2		
3	3	4		
4	3	5		
5	5	6		
6	6	7		
7	7	8		

	δ_H	δ_V	θ
1	0	0	1
2	0	0	0
8	1	0	1

3	4.5
6	2.0

5	2	0
6	0	10

Figure 4.2 Arrays associated with the definition of irregular frames

lengths are calculated. Referring to array MP( ), Fig. 4.2c, if a member reference number is defined by I then the joint reference number at each end of that member will be MP(I, 2) and MP(I, 3). From array JC( )—see Fig. 4.2b—the *x* and *y* coordinates of joint number MP(I, 2) are therefore JC(MP(I, 2), 2) and JC(MP(I, 2), 3) respectively. Similarly the coordinates of the other joint are JC(MP(I, 3), 2) and JC(MP(I, 3), 3). Note here that the *row* location of an element in one array (i.e. JC( )) is defined by the *content* of an element from another array (i.e. MP( )). Using these coordinates to define the ends of a member its length may therefore be calculated.

To complete the structural description a third array contains the support conditions. In Fig. 4.2d this array is called SC( ). The first column of the array holds the reference numbers of joints which constitute supports. The type of support involved will impose particular zero deformations; for example the support at joint number (1) is hinged and the deformations in both the *x* and *y* directions (δ_H and δ_V) will therefore be zero. These values are recorded in the second and third elements of the row. The numeral 1 in the fourth element of that row tells the computer that the support rotation is unknown. If NSU is the number of supports then array SC( ) will have NSU rows and 4 columns.

Further arrays are required to store the loading data. In this respect there will be as many arrays as there are basic load types. Two such arrays are shown in Fig. 4.2e where UDL( ) and JL( ) are concerned with transverse uniformly distributed loads and joint point loads respectively.

4.3 Structural Analysis

4.3.1 Introduction

The principles of structural behaviour which were established by 19th century applied mathematicians still form the basis of many contemporary methods of structural analysis. A direct application of these principles describes structural behaviour in a set of linear algebraic simultaneous equations which relate the structural geometry and displacements to the applied loads. For long after the principles were evolved the weight of arithmetic involved in the solution of the equations precluded their application from anything but simple structural forms. With the advent of electronic digital computers and their proficiency in executing repetitive arithmetic it was natural for analysts to reappraise those early ideas and to develop them in more sophisticated forms. And considering the nature of the problem it was inevitable that matrix algebra would be found to be a powerful shorthand with which to describe the overall behaviour of a structure and the interdependence of its parts. The strength of matrix methods of structural analysis lies with the fact that their use make it no longer necessary to catalogue structures into precise types or to apply a special method of analysis to a particular structural type. All structures belonging to the same category, however loaded, yield to the same treatment.

The traditional design office methods of slope-deflection and moment distribution, whilst stemming from the same roots as matrix methods, are only easily applied in the computer context to structures comprising a regular combination of rectangular cells. Even so, a large number of structures take this form and since moment distribution in particular (or preferably one of its derivatives, successive shear corrections) often leads to a more rapid solution and generally requires less computer storage space than do matrix methods, it can often be used to advantage.

The function of analysis is to furnish an assessment of the force actions and deformations induced by a load system. The necessarily iterative nature of the design of indeterminate structures means that throughout the initial design stages the structural properties are in a state of flux; accurate assessments of force actions are therefore only accurate in so far as they relate to the immediately previous assessment of section proportions. Whilst, because of program length, it is desirable to incorporate only one method of analysis in a design program a case can be made for the savings in computing time to be gained by arranging for successive analyses to produce, from a relatively coarse beginning, increasingly accurate results. With a relaxation type of solution this may be done by aranging for the criterion on which an analysis is terminated to become more severe as the number of overall design iterations increases. In the case of a stiffness matrix approach it might be profitable to reduce the number of equations to be solved by suppressing the secondary effects due to axial deformation during the initial design stages.

The expected relationship between the calculated force actions and those actually attained in practice should affect the choice of analytical method and the degree of accuracy to which it is pursued. Material behaviour, construction tolerances and the precision with which the loads have been estimated will also have been considered when making this decision. For structures deserving an 'accurate' treatment a two-level investigation may save computing time. At the lower level, design programs which are based upon acceptable approximate methods of analysis could be used to produce a number of rapid, but relatively unrefined solutions to the same problem for qualitative comparison; Program RCF12, Chapter 5, could be considered in this way. On the assumption that the relative merits of each solution were still valid even after refinement only one need then be chosen for a more thorough investigation.

In keeping with the general approach so far established of describing *familiar* problems in programming terms, the structural analysis programming which is described in the following sections is confined to a discussion of the slope-deflection method. It will be seen that this approach has much in common with stiffness matrix methods and should serve as a useful preliminary to further study.

4.3.2 An Outline of the Slope-Deflection Method

The slope-deflection equations:

$$M_{AB} = \frac{2EI}{L}\left(2\theta_A + \theta_B - \frac{3(\Delta_B - \Delta_A)}{L}\right) - M^F_{AB}$$

$$M_{BA} = \frac{2EI}{L}\left(\theta_A + 2\theta_B - \frac{3(\Delta_B - \Delta_A)}{L}\right) + M^F_{BA}$$

relate the bending moments generated at the ends of a member AB (see Fig. 4.3) to the rotations of its ends (θ_A and θ_B), the rotation of the member $(\Delta_B - \Delta_A)/L$ and the fixed end moments (M^F_{AB} and M^F_{BA}) due to transverse loading.

In a frame of the type shown in Fig. 4.4a the pattern of members and supports prevent vertical sway. If, therefore, all axial member deformations are neglected, then the *beams* will not rotate. Writing $K = EI/L$ the *beam* slope-deflection equations become:

$$\begin{aligned} M_{AB} &= 4K^B\theta_A + 2K^B\theta_B - M^F_{AB} \\ M_{BA} &= 2K^B\theta_A + 4K^B\theta_B + M^F_{BA} \end{aligned} \qquad (4.1)$$

If the columns are not subjected to transverse loads within their lengths then their slope-deflection equations will not include a fixed end moment term; but because horizontal sway is possible the *columns* may rotate. For *columns* the equations become:

$$\begin{aligned} M_{AB} &= 4K^C\theta_A + 2K^C\theta_B - \frac{6K^C\Delta_B}{L^C} + \frac{6K^C\Delta_A}{L^C} \\ M_{BA} &= 2K^C\theta_A + 4K^C\theta_B - \frac{6K^C\Delta_B}{L^C} + \frac{6K^C\Delta_A}{L^C} \end{aligned} \qquad (4.2)$$

For a frame with B-bays and S-storeys there will be $S*(B+1)$ joint rotations (θ) and S sway displacements (Δ). If axial member deformations are neglected then $S*(B+2)$ equations are sufficient to determine the unknown displacements. These equations are furnished by considering the moment equilibrium of each joint and the horizontal equilibrium at each floor level. Assuming that values of K^B, K^C, L^B, L^C, M^F_{AB}, M^F_{BA}, θ and Δ for the structure are stored in arrays dimensioned as shown in Fig. 4.4b, then the moment equilibrium equation for a joint specified by its (I, J) position in the structure may be written as:

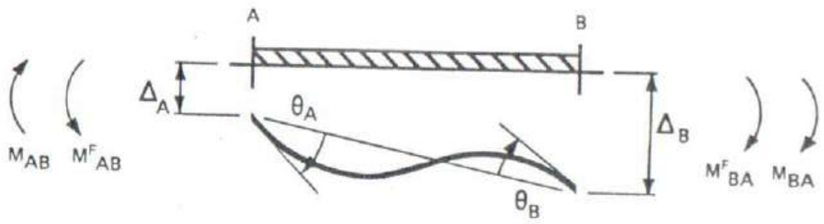


Figure 4.3 Bending moments generated at the ends of member AB

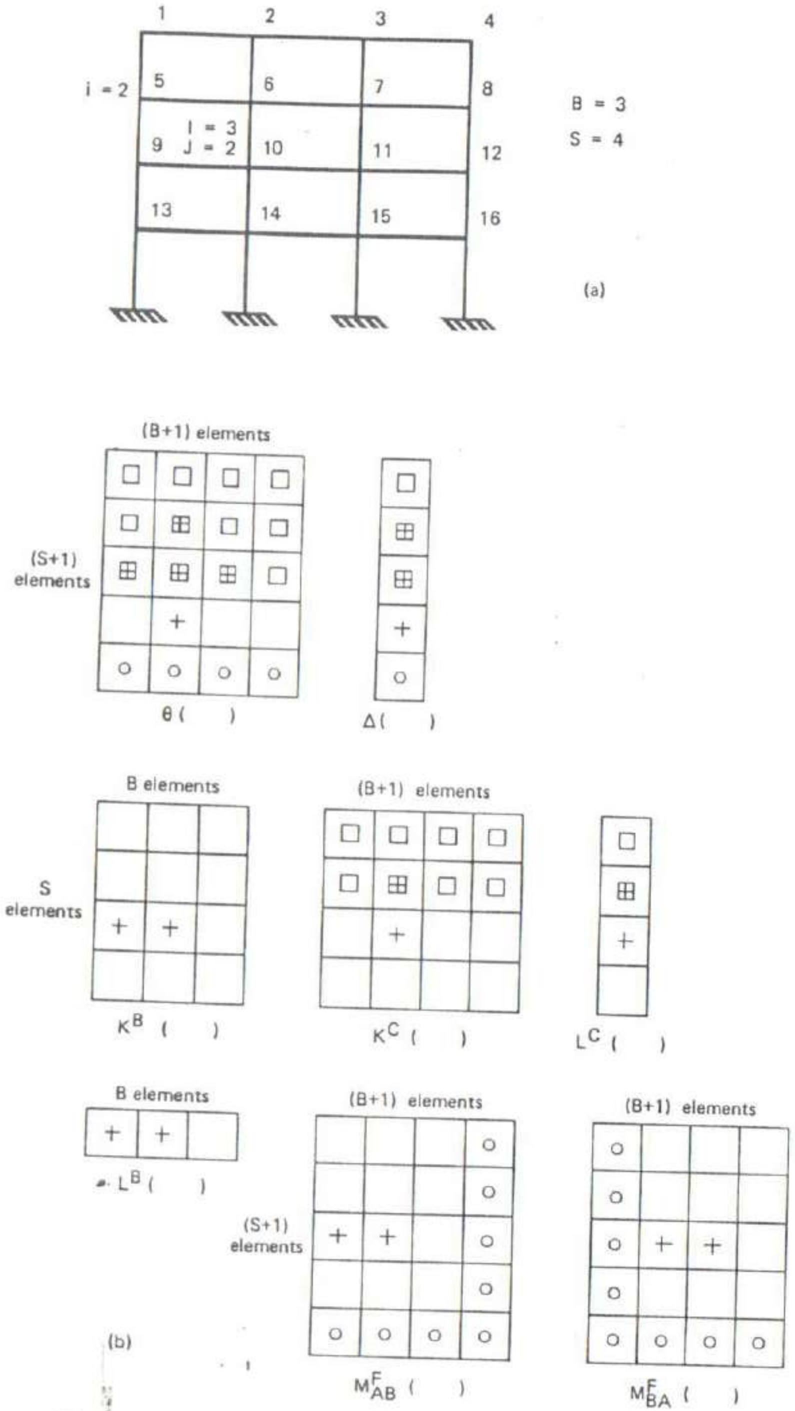


Figure 4.4 Arrays associated with the equilibrium equations

$$
\begin{aligned}
&4K^{B}_{I,J}\theta_{I,J} + 2K^{B}_{I,J}\theta_{I,J+1} + 4K^{B}_{I,J-1}\theta_{I,J} + 2K^{B}_{I,J-1}\theta_{I,J-1} + 4K^{C}_{I-1,J}\theta_{I,J} \\
&+ 2K^{C}_{I-1,J}\theta_{I-1,J} - 6K^{C}_{I-1,J}(\Delta_{I-1} - \Delta_{I})/L^{C}_{I-1} + 4K^{C}_{I,J}\theta_{I,J} + 2K^{C}_{I,J}\theta_{I+1,J} \\
&- 6K^{C}_{I,J}(\Delta_{I} - \Delta_{I+1})/L^{C}_{I} = -(M^{F}_{AB(I,J)} - M^{F}_{BA(I,J-1)})
\end{aligned}
\tag{4.3}
$$

This states that the sum of the moments at the ends of members meeting at a joint is equal and opposite to the out-of-balance moment acting at that joint.

The equation which defines horizontal equilibrium at a floor level which is identified by *i* is:

$$
\begin{aligned}
&\sum_{J=1}^{J=B+1}\left(\frac{6K^{C}_{i-1,J}}{L^{C}_{i-1}}\left(\theta_{i-1,J} + \theta_{i,J} - \frac{2}{L^{C}_{i-1}}(\Delta_{i-1} - \Delta_{i})\right)\right. \\
&\qquad\left. - \frac{6K^{C}_{i,J}}{L^{C}_{i}}(\theta_{i,J} + \theta_{i+1,J} - \frac{2}{L^{C}_{i}}(\Delta_{i} - \Delta_{i+1})\right) \\
&= \text{The horizontal load applied at floor level}
\end{aligned}
\tag{4.4}
$$

The LHS of this equation represents the sum of the column shear forces at a given floor level—each shear force is the result of dividing the sum of the moments acting at the end of a column by its length.

In Fig. 4.4b the array elements involved in equation (4.3) are marked thus +, whilst those in equation (4.4) are marked □.

The 3-bay, 4-storey frame shown in Fig. 4.4a may be analysed by solving the 20 simultaneous linear algebraic equations resulting from 16 joint equilibrium conditions (the column/foundation connections are not considered here because their displacements are known to be zero), and 4 horizontal equilibrium conditions.

When the joints are numbered (along rows) from 1 to 20, with the first joint located at I = J = 1, the relationship between a joint reference number and its (I, J) location is:

JOINT REFERENCE NUMBER = (I − 1)∗(B + 1) + J

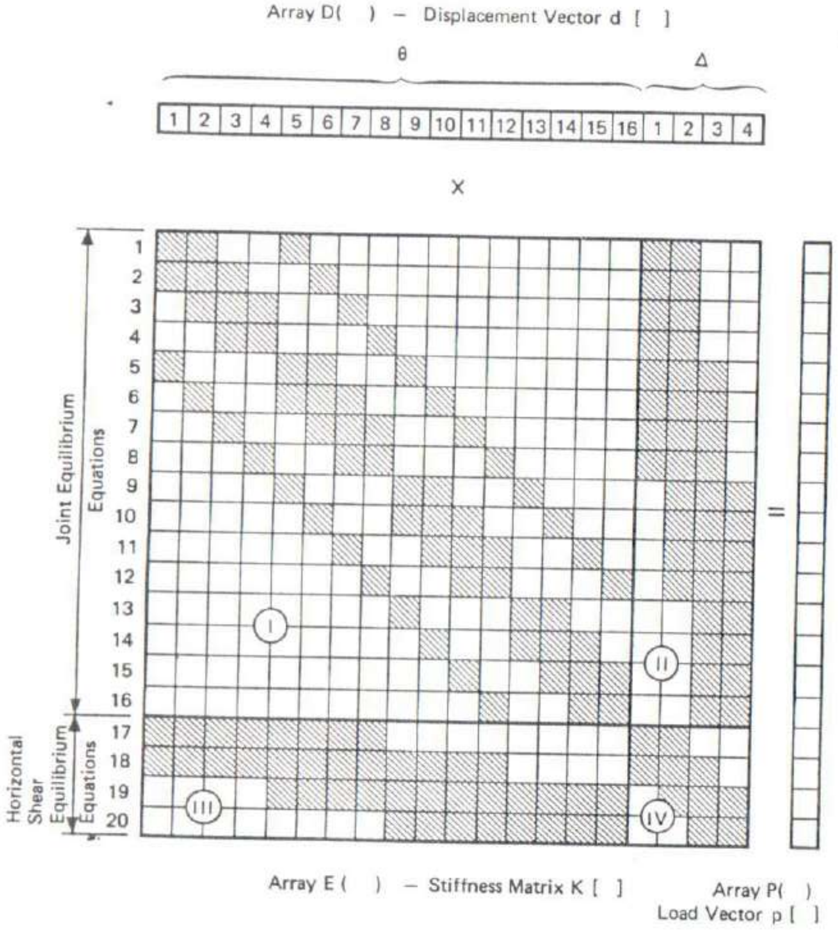
If the first 16 equations are set up in the order of their joint reference numbers, then for (I = 3, J = 2)—see Fig. 4.4a—the joint reference number (and hence the equation number) will be (3 − 1)∗(3 + 1) + 2 = 10. Thus LHS of equation number (10) becomes:

$$
\begin{aligned}
&4(K^{B}_{3,2} + K^{B}_{3,1} + K^{C}_{2,2} + K^{C}_{3,2})\theta_{10} + 2K^{B}_{3,2}\theta_{11} + 2K^{B}_{3,1}\theta_{9} + 2K^{C}_{2,2}\theta_{6} \\
&+ 2K^{C}_{3,2}\theta_{14} - \frac{6K^{C}_{2,2}\Delta_{2}}{L^{C}_{2}} + 6\left(\frac{K^{C}_{2,2}}{L^{C}_{2}} - \frac{K^{C}_{3,2}}{L^{C}_{3}}\right)\Delta_{3} - \frac{6K^{C}_{3,2}\Delta_{4}}{L^{C}_{3}}
\end{aligned}
$$

The last four equations are set up by considering the floors in order, numbered consecutively from the top of the frame. For *i* = 2 the LHS of equation number (18) takes the form:

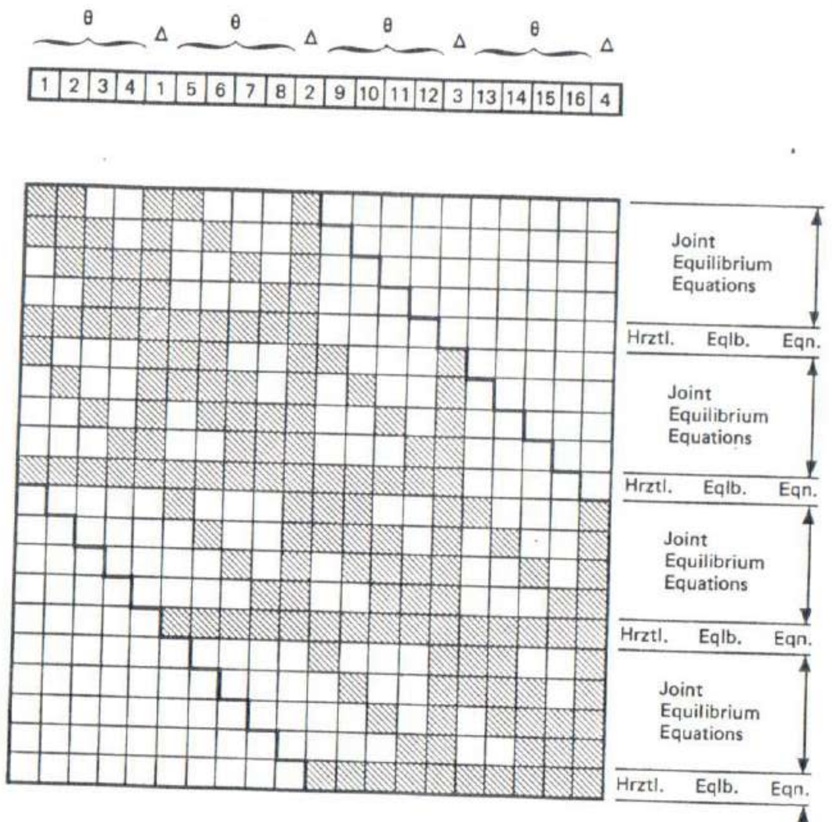
$$\sum_{J=1}^{J=4}\left(\frac{6K^{C}_{1,J}\theta_{1,J}}{L^{C}_{1}}+\left(\frac{6K^{C}_{1,J}}{L^{C}_{1}}+\frac{6K^{C}_{2,J}}{L^{C}_{2}}\right)\theta_{2,J}-\frac{6K^{C}_{2,J}\theta_{3,J}}{L^{C}_{2}}\right.$$
$$\left.-\frac{12K^{C}_{1,J}\Delta_{1}}{(L^{C}_{1})^{2}}+\left(\frac{12K^{C}_{1,J}}{(L^{C}_{1})^{2}}+\frac{12K^{C}_{2,J}}{(L^{C}_{2})^{2}}\right)\Delta_{2}-\frac{12K^{C}_{2,J}\Delta_{3}}{(L^{C}_{2})^{2}}\right)$$

It will be appreciated that, in effect, the LHS of each equation also contains zero coefficients to match the displacements not involved in that particular equation of equilibrium. The pattern of displacement coefficients for the 20 equations is shown in array E( ), Fig. 4.5, where zero coefficients are shown as blank elements and non-zero coefficients are hatched. A notable feature of the array is the symmetry of its contents about the leading diagonal. Later in this section it is shown how this property may be exploited.



The distribution of non-zero elements in array P( ) depends upon the loading pattern

Figure 4.5 Pattern of displacement coefficients



(a) Modified Array E(

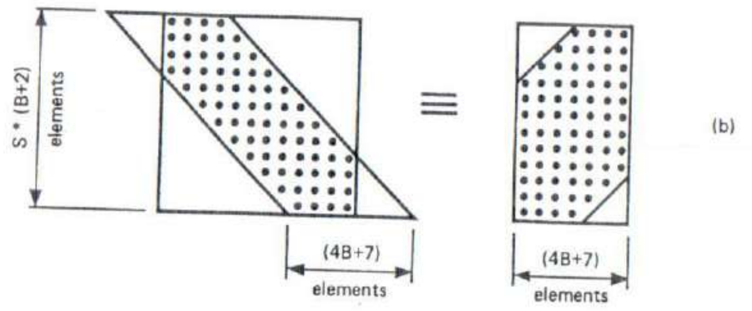


Figure 4.6 Displacement coefficients concentrated within a diagonal band

The content of each non-zero element in the array is a function of member properties. The distribution of displacement coefficients throughout the array reflects the way in which the members of the structure are connected to each other. Array E( ) represents the *Frame Stiffness Matrix* (K) for the particular case of a frame with members of infinitely large cross-sectional area, i.e. it is assumed that the axial displacement between joints is zero.

To pursue this analogy with matrix methods a little further, array D( )—see Fig. 4.5—which represents the unknown displacements, is equivalent to the *Displacement Vector* (d); array P( ), which contains the RHS's of the equilibrium equations, is the *Load Vector* (p). In this example array P( ) has 20 elements. The contents of the first 16 elements are the RHS's of equation type (4.3) and the four remaining elements represent the RHS's of equation type (4.4). The set of equations may be described by the single equation p = Kd. A method of solving this set of equations is described in the next section.

By re-arranging the order of the terms in the equations and the order in which the equations are derived it is possible, whilst still preserving symmetry, to concentrate all the non-zero coefficients within a diagonal band of elements, the 'bandwidth' comprising a decreasing proportion of the total number of displacements as the number of storeys increases. Fig. 4.6a shows such an arrangement of equations for the frame under discussion. When arranged in this way the non-zero coefficients are embraced by a 'bandwidth' of (4B + 7) elements and all the coefficients outwith this band are zero. By storing only those coefficients which fall within the diagonal band it is possible to replace the $(S(B+2))^2$ element array with one containing S(4B + 7) (B + 2) elements —see Fig. 4.6b. In the case of large multi-storey structures the consequent reduction in the array size needed to hold the stiffness matrix makes substantial savings in the computer storage requirements. A further useful reduction in the storage requirements for this array can be made by recognising that, since the coefficients within the band are symmetrical about the leading diagonal, the content of the elements on and above the leading diagonal wholly reflect the properties of the structure; only those elements therefore need to be stored.

These devices naturally affect the way in which array elements are manipulated during the solution of the equations. In the ensuing discussion the total stiffness matrix is considered and the refinements mentioned above are ignored.

4.3.3 Solving the Equations

A simple and orderly way of solving a set of simultaneous equations is to use the method of successive elimination of variables. The method will be discussed in relation to a set of n simultaneous equations, where $x_1 \ldots\ldots\ldots x_n$ are the variables to be determined:

$$\begin{array}{l}
x_1 \quad x_2 \quad x_3 \ldots\ldots\ldots\ldots\ x_n \\
\hline
a_{11} + a_{12} + a_{13} + \ldots\ldots\ldots + a_{1n} = C_1 \qquad (1) \\
a_{21} + a_{22} + a_{23} + \ldots\ldots\ldots + a_{2n} = C_2 \qquad (2) \\
a_{31} + a_{32} + a_{33} + \ldots\ldots\ldots + a_{3n} = C_3 \qquad (3) \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
a_{n1} + a_{n2} + a_{n3} + \ldots\ldots\ldots + a_{nn} = C_n \qquad (n)
\end{array} \qquad (4.5)$$

The first object is to eliminate x_1 from the second and subsequent equations. This is done in two stages:

1. Assuming that $a_{11}, a_{21}, \ldots\ldots a_{n1} \neq 0$ then each equation is divided by its coefficient of x_1. The effect of this is to make all the non-zero coefficients in the first column equal to 1;
2. The first equation is then subtracted from each equation in turn. Thus the first coefficient in the modified equations (2)...... (*n*) becomes zero.

The set of equations (4.5) therefore takes the form:

$$\begin{array}{l}
x_1 \quad x_2 \quad x_3 \ldots\ldots\ldots\ldots\ x_n \\
\hline
1 + a'_{12} + a'_{13} + \ldots\ldots\ldots + a'_{1n} = C'_1 \qquad (1) \\
0 + a'_{22} + a'_{23} + \ldots\ldots\ldots + a'_{2n} = C'_2 \qquad (2) \\
0 + a'_{32} + a'_{33} + \ldots\ldots\ldots + a'_{3n} = C'_3 \qquad (3) \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
0 + a'_{n2} + a'_{n3} + \ldots\ldots\ldots + a'_{nn} = C'_n \qquad (n)
\end{array} \qquad (4.6)$$

where the primes indicate that the coefficients are now modified as a result of the arithmetical operations which have just been carried out. In the set of equations (4.5) equation (1) is called the *pivotal equation* and the first element in that equation is the *pivotal element*.

By applying the same procedure to the equations (2) ..... (n) in the modified set (4.6), with equation (2) as the new pivotal equation and a'_{22} as the pivotal element, x_2 is eliminated from the equations (3) to (*n*). A further $(n-2)$ similar operations carried out on successively smaller blocks of equations converts the original set (4.5) to the form:

$$\begin{array}{l}
x_1 \quad x_2 \quad x_3 \ldots\ldots\ldots\ldots\ x_n \\
\hline
1 + a'_{12} + a'_{13} + \ldots\ldots\ldots + a'_{1n} = C'_1 \qquad (1) \\
0 + 1 \quad + a''_{23} + \ldots\ldots\ldots + a''_{2n} = C''_2 \qquad (2)
\end{array}$$

$$\begin{array}{lr} 0 \ + 0 \quad + 1 \quad + \ldots\ldots\ldots + a'''_{3n} = C'''_3 & (3) \\ \cdot & \\ \cdot & \\ \cdot & \\ \cdot & \\ \cdot & \\ 0 \ + 0 \quad + 0 \quad + \ldots\ldots\ldots + 1 \quad = C_n^{('n)} & (n) \end{array} \qquad (4.7)$$

Equation (n) of the set (4.7) shows that $x_n = C_n^{('n)}$. The value of the variable x_{n-1} is calculated by substituting the known value of x_n into equation $(n-1)$. The remaining variables are determined in a similar way by substituting the values of known variables into each equation in turn.

Excepting for errors arising from 'rounding-off' calculated quantities the method described is an exact one which readily lends itself to computer programming techniques. The load-displacement equations which describe the behaviour of multi-cell structures are almost invariably well conditioned and can be expected to yield 'accurate' values for the displacements. However, one modification to the standard method, which requires little extra programming effort, can reduce the effect of a possible source of inaccuracy.

According to the method described above the pivotal equation was defined as being the first equation in the block of equations waiting to be processed. But if the pivotal element is numerically small in comparison with the remaining elements of the column which it occupies, then the modified coefficients in the pivotal equation will probably be large in comparison with the quantities from which they will eventually be subtracted. This situation is a potential source of accumulative errors which may be largely rectified by 'exchanging pivots'. Before the process of eliminating a variable is begun the location is found of the numerically largest coefficient in the first column of elements of the remaining block. The equation which is defined by the location of this coefficient is then chosen as the pivotal equation and exchanged with the first equation in this block. By choosing the pivotal equation in this way, its effect on the remaining equations will be minimal.

The basic solution is complete when the displacements have been determined. The moments at the ends of each member are calculated by substituting the relevant loads and displacements into equations (4.1) and (4.2).

4.3.4 Program Specification—SD1

SD1 is a structural analysis program which is based upon the slope-deflection method. Its application is confined to the analysis of multi-bay, multi-storey frames having zero displacements (both rotational and linear) at foundation level. The load-displacement equations are solved by the method of successive elimination of variables. Each analysis is limited to a single load condition which may comprise any chosen pattern of uniformly distributed beam loads and/or horizontal loads applied at floor levels. The results output comprises a list of member end-moments and joint displacements.

Even within the limitations imposed by the single structural type, an expansion of the program area which deals with the calculation of force actions to include the shear and axial forces and the maximum span bending moments would make SD1 a more useful design tool.

4.3.5 The SD1 Flow Diagram

The flow diagram for this program is shown in Fig. 4.7. Because its form is substantially linear the flow diagram could have been replaced by an ordered list of essential program areas. The only major loops in the program are those introduced to enable further problems to be solved during a single program run.

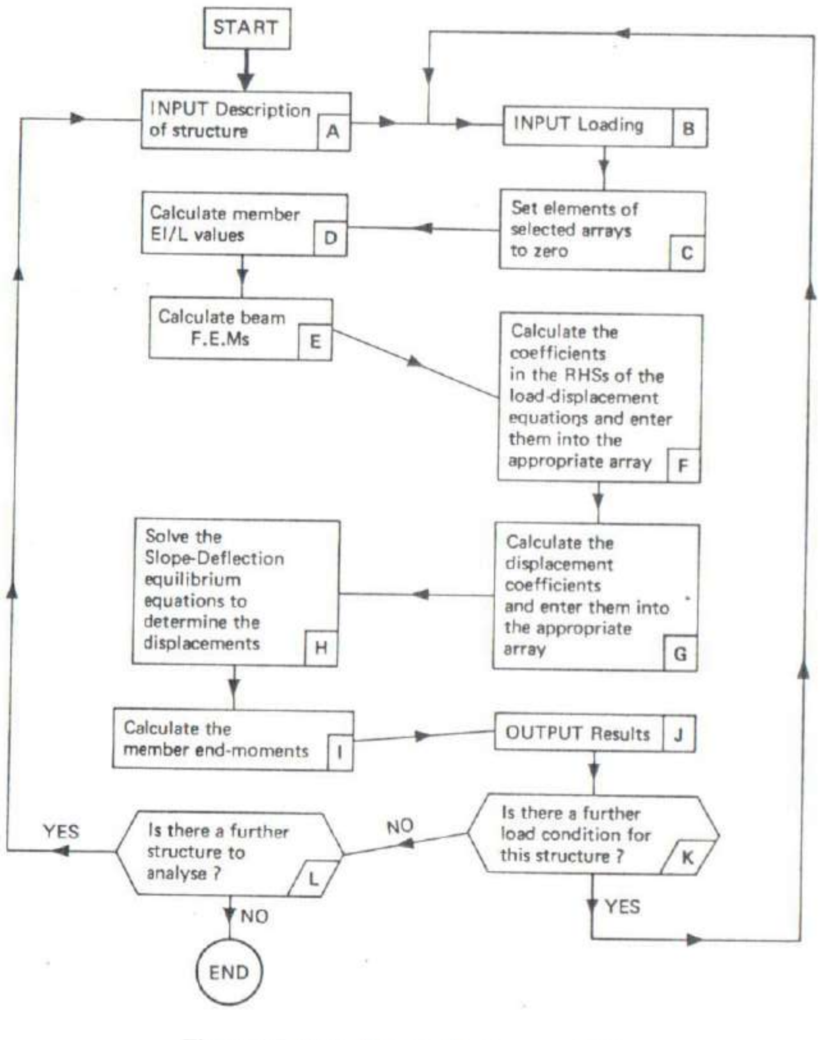


Figure 4.7 Flow diagram for program SD1

The input of structural and loading parameters are treated in separate blocks (A and B) so that further load conditions may be examined without the necessity for repeatedly reading the structural geometry into the computer. At Block (C) certain arrays are formally cleared to ensure that the state of their contents is in no doubt when they come to be used. Blocks (D), (E), (F), (G), (H) and (I) represent the main steps to be followed in an analysis. The calculation of member EI/L values and fixed end-moments at Blocks (D) and (E) clears the way for setting up the load-displacement equations at Blocks (F) and (G). Following the solution of these equations at Block (H) the force actions are determined at Block (I).

4.3.6 Description of Program SD1

A list of the variables and arrays used in this program is given below.

A	Row counter
B	Number of bays
C	Load case trigger
D	Further problem trigger
E( )	Contains the displacement coefficients
E	Young's modulus
F	Frame reference number
G	Temporary store for use during an exchange
H	Array dimension
I	Row counter
J	Column counter
K	Row number locating new pivot
L( )	Contains the beam second moments of area
M( )	Contains the column second moments of area
N	Counts a number of operations
O	Locates a row in E( )
P( )	Initially contains the RHS's of the load-displacement equations, finally the displacements
P	Locates a column in E( )
Q( )	Contains the North moments
Q	Locates a column in E( )
R( )	Contains the East moments
R	Locates a column in E( )
S( )	Contains the South moments
S	Number of storeys
T( )	Contains the West moments
T	Locates a column in E( )
U( )	Contains beam UDL's
V( )	Contains wind loads
W( )	Contains column centres (i.e. beam spans)
X( )	Contains storey heights (i.e. column lengths)
Y( )	Contains beam stiffnesses
Z( )	Contains column stiffnesses
Z	Temporary store for use during an exchange

Program SD1 is listed below and should be read in conjunction with the flow diagram shown in Fig. 4.7.

```
100 DIM E(35,35)
110 DIM P(35)
120 PRINT "INPUT FRAME REFERENCE NUMBER";
130 INPUT F
140 PRINT
150 PRINT "ANALYSIS OF FRAME NUMBER"F"BY SLOPE-DEFLECTION METHOD"
160 PRINT
```

127

```
170 PRINT "INPUT NUMBER OF BAYS AND STOREYS";
180 INPUT B,
190 PRINT
200 PRINT "INPUT COLUMN CENTRES - (M)"
210 FOR J=1 TO B
220 INPUT W(J)
230 NEXT J
240 PRINT
250 PRINT "INPUT STOREY HEIGHTS - (M)"
260 FOR I=1 TO S
270 INPUT X(I)
280 NEXT I
290 PRINT
300 PRINT "INPUT BEAM I-VALUES - (MM+4)"
310 FOR I=1 TO S
320 FOR J=1 TO B
330 INPUT L(I,J)
340 NEXT J
350 NEXT I
360 PRINT
370 PRINT "INPUT COLUMN I-VALUES - (MM+4)"
380 FOR I=1 TO S
390 FOR J=1 TO B+1
400 INPUT M(I,J)
410 NEXT J
420 NEXT I
430 PRINT
440 PRINT "INPUT YOUNG'S MODULUS - (N/MM+2)"
450 INPUT E
460 PRINT
470 PRINT "INPUT BEAM UDL'S - (KN/M)"
480 FOR I=1 TO S
490 FOR J=1 TO B
500 INPUT U(I,J)
510 NEXT J
520 NEXTI
530 PRINT
540 PRINT "INPUT WIND LOADS - (KN)"
550 FOR I=1 TO S
560 INPUT V(I)
570 NEXT I
580 PRINT
590 FOR I=1 TO S*(B+2)
600 FOR J=1 TO S*(B+2)
610 LET E(I,J)=0
620 NEXT J
630 NEXT I
640 FOR I=1 TO S*(B+2)
650 LET P(I)=0
660 NEXT I
670 FOR I=1 TO S+1
680 FOR J=1 TO B+1
690 LET O(I,J)=0
700 LET S(I,J)=0
710 LET R(I,J)=0
720 LET T(I,J)=0
730 NEXT J
740 NEXT I
750 FOR I=1 TO S
760 FOR J=1 TO B
770 LET Y(I,J)=E*L(I,J)/(W(J)*1E9)
780 NEXT J
790 NEXT I
800 FOR I=1 TO S
810 FOR J=1 TO B+1
820 LET Z(I,J)=E*M(I,J)/(X(I)*1E9)
```

```
830 NEXT J
840 NEXT I
850 FOR I=1 TO S
860 FOR J=1 TO B
870 LET T(I,J+1)=U(I,J)*W(J)↑2/12
880 LET R(I,J)=-T(I,J+1)
890 NEXT J
900 NEXT I
910 FOR I=1 TO S
920 FOR J=1 TO B+1
930 LET O=(I-1)*(B+1)+J
940 LET P(O)=-(T(I,J)+R(I,J))
950 NEXT J
960 NEXT I
970 FOR I=1 TO S
980 LET P(S*(B+1)+I)=V(I)
990 NEXT I
1000 FOR I=1 TO S
1010 FOR J=1 TO B+1
1020 LET O=(I-1)*(B+1)+J
1030 LET P=O+1
1040 LET R=O-1
1050 LET Q=(I-2)*(B+1)+J
1060 LET T=I*(B+1)+J
1070 IF I=1 THEN 1100
1080 LET E(O,O)=E(O,O)+4*Z(I-1,J)
1090 LET E(O,Q)=2*Z(I-1,J)
1100 LET E(O,O)=E(O,O)+4*Z(I,J)
1110 IF I=S THEN 1130
1120 LET E(O,T)=2*Z(I,J)
1130 IF J=B+1 THEN 1170
1140 LET E(O,O)=E(O,O)+4*Y(I,J)
1150 LET E(O,P)=2*Y(I,J)
1160 IF J=1 THEN 1190
1170 LET E(O,O)=E(O,O)+4*Y(I,J-1)
1180 LET E(O,R)=2*Y(I,J-1)
1190 NEXT J
1200 NEXT I
1210 FOR I=1 TO S
1220 FOR J=1 TO B+1
1230 LET O=(I-1)*(B+1)+J
1240 LET P=(B+1)*S+I-1
1250 LET Q=P+1
1260 LET R=Q+1
1270 IF I>1 THEN 1300
1280 LET E(O,Q)=-6*Z(I,J)/X(I)
1290 GOTO 1330
1300 LET E(O,Q)=6*Z(I-1,J)/X(I-1)-6*Z(I,J)/X(I)
1310 LET E(O,P)=-6*Z(I-1,J)/X(I-1)
1320 IF I=S THEN 1340
1330 LET E(O,R)=6*Z(I,J)/X(I)
1340 NEXT J
1350 NEXT I
1360 FOR I=1 TO S
1370 FOR J=1 TO B+1
1380 LET O=S*(B+1)+I
1390 LET P=(I-1)*(B+1)+J
1400 LET Q=P+B+1
1410 LET R=P-B-1
1420 LET E(O,P)=-6*Z(I,J)/X(I)
1430 IF I=1 THEN 1470
1440 LET E(O,P)=E(O,P)+6*Z(I-1,J)/X(I-1)
1450 LET E(O,R)=6*Z(I-1,J)/X(I-1)
1460 IF I=S THEN 1480
1470 LET E(O,Q)=-6*Z(I,J)/X(I)
1480 NEXT J
```

```
1490 NEXT I
1500 FOR I=1 [illegible] S
1510 LET O=S[illegible]B+1)+I
1520 IF I=1 THEN 1580
1530 FOR J=1 TO B+1
1540 LET E(O,O)=E(O,O)+12*Z(I,J)/X(I)↑2+12*Z(I-1,J)/X(I-1)↑2
1550 LET E(O,O-1)=E(O,O-1)-12*Z(I-1,J)/X(I-1)↑2
1560 NEXT J
1570 GOTO 1610
1580 FOR J=1 TO B+1
1590 LET E(O,O)=E(O,O)+12*Z(I,J)/X(I)↑2
1600 NEXT J
1610 IF I=S THEN 1650
1620 FOR J=1 TO B+1
1630 LET E(O,O+1)=E(O,O+1)-12*Z(I,J)/X(I)↑2
1640 NEXT J
1650 NEXT I
1660 LET H=S*(B+2)
1670 FOR N=1 TO H
1680 LET K=0
1690 FOR I=N TO H
1700 IF ABS(E(I,N))<K THEN 1730
1710 LET K=ABS(E(I,N))
1720 LET L=I
1730 NEXT I
1740 FOR J=N TO H
1750 LET G=E(L,J)
1760 LET E(L,J)=E(N,J)
1770 LET E(N,J)=G
1780 NEXT J
1790 LET X=P(L)
1800 LET P(L)=P(N)
1810 LET P(N)=X
1820 FOR I=N TO H
1830 IF E(I N)=0 THEN 1890
1840 LET Z=E(I,N)
1850 FOR J=N TO H
1860 LET E(I,J)=E(I,J)/Z
1870 NEXT J
1880 LET P(I)=P(I)/Z
1890 NEXT I
1900 IF N=H THEN 1990
1910 FOR I=N+1 TO H
1920 IF E(I,N)=0 THEN 1970
1930 FOR J=N TO H
1940 LET E(I,J)=E(I,J)-E(N,J)
1950 NEXT J
1960 LET P(I)=P(I)-P(N)
1970 NEXT I
1980 NEXT N
1990 FOR A=2 TO H
2000 LET I=H-A+1
2010 FOR J=I+1 TO H
2020 LET P(I)=P(I)-E(I,J)*P(J)
2030 NEXT J
2040 NEXT A
2050 FOR I=1 TO S
2060 FOR J=1 TO B
2070 LET R(I,J)=R(I,J)+4*Y(I,J)*P((I-1)*(B+1)+J)
2080 LET R(I,J)=R(I,J)+2*Y(I,J)*P((I-1)*(B+1)+J+1)
2090 LET T(I,J+1)=T(I,J+1)+2*Y(I,J)*P((I-1)*(B+1)+J)
2100 LET T(I,J+1)=T(I,J+1)+4*Y(I,J)*P((I-1)*(B+1)+J+1)
2110 NEXT J
2120 NEXT I
2130 FOR I=1 TO S
2140 FOR J=1 TO B+1
```

```
2150 IF I=S THEN 2200
2160 LET Q(I+1,J)=4*Z(I,J)*P(I*(B+1)+J)
2170 LET Q(I+1,J)=Q(I+1,J)+2*Z(I,J)*P((I-1)*(B+1)+J)
2180 LET Q(I+1,J)=Q(I+1,J)-6*Z(I,J)*(P(S*(B+1)+I))/X(I)
2181 LET Q(I+1,J)=Q(I+1,J)+6*Z(I,J)*P(S*(B+1)+I+1)/X(I)
2190 GOTO 2230
2200 LET Q(I+1,J)=2*Z(I,J)*P((I-1)*(B+1)+J)
2210 LET Q(I+1,J)=Q(I+1,J)-6*Z(I,J)*P(S*(B+1)+I)/X(I)
2220 GOTO 2260
2230 LET S(I,J)=2*Z(I,J)*P(I*(B+1)+J)+4*Z(I,J)*P((I-1)*(B+1)+J)
2240 LET S(I,J)=S(I,J)-6*Z(I,J)*(P(S*(B+1)+I)-P(S*(B+1)+I+1))/X(I)
2250 GOTO 2270
2260 LET S(I,J)=4*Z(I,J)*P((I-1)*(B+1)+J)-6*Z(I,J)*P(S*(B+1)+I)/X(I)
2270 NEXT J
2280 NEXT I
2290 PRINT
2300 PRINT "MEMBER END MOMENTS (KNM)"
2310 PRINT "(MEMBERS INTERSECT AT JOINTS DEFINED"
2320 PRINT " BY THEIR I,J LOCATIONS.)"
2330 PRINT
2340 PRINT "        I  J     NM          SM          EM          WM"
2350 FOR I=1 TO S+1
2360 FOR J=1 TO B+1
2370 PRINT "JOINT"I;J,Q(I,J),S(I,J),R(I,J),T(I,J)
2380 NEXT J
2390 NEXT I
2400 PRINT
2410 PRINT
2420 PRINT "DISPLACEMENTS"
2430 PRINT
2440 PRINT "JOINT ROTATIONS (RADIANS)"
2450 PRINT "        I  J"
2460 FOR I=1 TO S+1
2470 FOR J=1 TO B+1
2480 IF I=S+1 THEN 2510
2490 PRINT "JOINT"I;J,P((I-1)*(B+1)+J)
2500 GOTO 2520
2510 PRINT "JOINT"I;J," 0"
2520 NEXT J
2530 NEXT I
2540 PRINT
2550 PRINT
2560 PRINT "HORIZONTAL DISPLACEMENTS"
2570 PRINT
2580 PRINT "                                   DISPLACEMENT"
2590 FOR I=1 TO S+1
2600 IF I=S+1 THEN 2660
2610 IF I>1 THEN 2640
2620 PRINT "       ROOF LEVEL",P(S*(B+1)+I)*1000"MM"
2630 GOTO 2670
2640 PRINT "   FLOOR"S-I+1"LEVEL",P(S*(B+1)+I)*1000"MM"
2650 GOTO 2670
2660 PRINT "FOUNDATION LEVEL                0 MM"
2670 NEXT I
2680 PRINT
2690 PRINT
2700 PRINT "IF FURTHER LOAD CASES FOR THIS FRAME TYPE 1 ELSE 0";
2710 INPUT C
2720 IF C=1 THEN 460
2730 FOR I=1 TO 10
2740 PRINT
2750 NEXT I
2760 PRINT "IF FURTHER FRAMES TO BE ANALYSED TYPE 1 ELSE 0";
2770 INPUT D
2780 IF D=1 THEN 120
2790 END
```

4.3.6.1 Blocks (A) and (B): Statements 100–580

To increase the capacity of the program for solving simultaneous equations the dimensions of E( ) and P( ), the largest arrays in the program, are arbitrarily set to 35 at lines 100 and 110. In terms of B and S (the number of bays and storeys) the number of equations required for the solution of a given frame is $S*(B+2)$. If the dimensions of the second largest arrays are not increased beyond that which is automatically allocated to them then for a solution by this program to be possible the expressions $(B+1) \ngtr 10$, $(S+1) \ngtr 10$ and $S*(B+2) \ngtr 35$ set the upper limits to the number of bays and storeys that can be handled in a specific case.

The remainder of Blocks (A) and (B) is wholly concerned with the input of structural parameters and loading. In comparison with the PRINT and INPUT statements which are included in Program RCF12 (Chapter 5) the standard to which SD1 requests information should be considered to be the minimum acceptable.

4.3.6.2 Block (C): Statements 590–740

Block (C) is an essential preliminary to the main calculation. The need to clear an array (i.e. to set the contents of its elements to zero) often arises if the array plays a role in the solution of more than one problem during the same program run. If, for example, quantities are entered into an array according to the statement:

96 LET C(I, J) = C(I, J) + K

then unless C(I, J) is set to zero before the next solution it will retain a non-zero value and cause arithmetical errors in subsequent calculations. With this kind of pitfall in mind the arrays E( ), P( ), Q( ), R( ), S( ) and T( ) are all cleared.

4.3.6.3 Block (D): Statements 750–840

By calculating and storing the K = EI/L value for each member at this stage the form of the statements which govern the calculation of the displacement coefficients at Block (G) is simplified. The beam span or column length used to determine K is inferred from the column centres or storey heights which were read into arrays L( ) and M( ) at Block (A).

4.3.6.4 Block (E): Statements 850–900

The arrays R( ) and T( ) store the bending moments which are induced at the left and right hand supports respectively. At this stage in the calculation they are the fixed end moments due to uniformly distributed span loads—see lines 850 to 900. Subsequently, at Block (I), these arrays record the final support moments after account has been taken of the effects of sway and joint rotation.

4.3.6.5 Block (F): Statements 910–990

Array P( ) has a dual role. At the beginning of an analysis its elements represent the RHS's of the load-displacement equations (the load vector); finally it contains the displacements (the displacement vector).

The load vector elements are defined by the RHS's of equations (4.3) and (4.4). The first S∗(B + 1) elements of P( ) contain − 1∗(the out of balance moment) at each joint. This quantity, calculated at line 940, is related to the correct element in P( ) by recognizing that a joint reference number and the number of its equilibrium equation are identical. If the location of a joint in the frame is specified by its (I, J) values, then the reference number of that joint is O = (I − 1)∗(B + 1) + J (see line 930). Thus P(O) is the location of the correct load vector element for joint number 0.

The remaining S elements of P( ) accommodate the RHS's of the horizontal shear equilibrium equations, i.e. the contents of array V( ). By assigning a value of I to each floor level in turn (beginning with I = 1 at roof level) the reference numbers for this last group of equations become S∗(B + 1) + I (see line 980).

4.3.6.6 Block (G): Statements 1000–1650

All the elements in array E( ) were set to zero at Block (C). The purpose of Block (G) is to calculate the non-zero displacement coefficients, as defined by equations (4.3) and (4.4), and to assign them to their correct locations in E( ). Array E( ) can then be thought of either as representing the displacement coefficients in the equations of equilibrium or the stiffness matrix for the frame.

The horizontal and vertical lines which partition array E( ) in Fig. 4.5 delineate four regions, each of which contains a distinctive pattern of coefficients. In the program a separate set of statements is used to insert the appropriate coefficients into each region.

These are:
I Statements 1000 to 1200
II Statements 1210 to 1350
III Statements 1360 to 1490
and IV Statements 1500 to 1650

The form taken by each of these sets of statements follows naturally from the structure of the relevant equation of equilibrium. The statements which govern the pattern of coefficients in Region I will be considered in detail. It is left to the reader to confirm that the statements governing the remaining regions are valid.

Region I will always have as many rows and columns as there are 'free' joints in the frame. We are therefore concerned with S∗(B + 1) equations of joint equilibrium and the same number of rotational displacements. Equation (4.3) shows that, in terms of the joint rotations, equilibrium is a function of the rotation ($\theta_{I,J}$) of the nominated joint and the rotations ($\theta_{I,J+1}$, $\theta_{I,J-1}$, $\theta_{I+1,J}$ and $\theta_{I-1,J}$) of the joints at the remote ends of the connected members. Thus, in general, a joint equilibrium equation will include five non-zero coefficients of θ. The reference number of an equation within Region I, depending as it does on the (I,J) location of the joint, is defined by 0 = (I − 1)∗(B + 1) + J at line 1020. O is also the location of $\theta_{I,J}$ in the displacement vector. Hence E(O,O) must contain the coefficient of $\theta_{I,J}$. P, R, Q and T, calculated at lines 1030 to 1060, are the displacement vector locations (and therefore the column locations in array E( )) of $\theta_{I,J+1}$, $\theta_{I,J-1}$, $\theta_{I-1,J}$ and $\theta_{I+1,J}$ respectively. The elements E(O, P), E(O, R), E(O, Q) and E(O, T) therefore contain the coefficients of these rotational displacements. It follows from equation (4.3) that at an internal joint which connects four members: E(O, O) = 4∗(the sum of the K-values of members connected by the joint) and E(O, P), E(O, R), E(O, Q) and E(O, T) = 2∗(the K-value of the relevant connected member), where K is stored appropriately in either Y( ) or Z( ) according to whether the member is a beam or a column.

For each joint in succession the contents of elements E(O, O), E(O, P), etc. are calculated at lines 1080 to 1180. The conditional statements at lines 1070, 1130 and 1160 take account of the fact that peripheral joints connect fewer than four members. The statement at line 1110 recognizes that since full fixity is assumed at foundation level, E(O, T) does not exist when I = S.

4.3.6.7 Block (H): Statements 1660–2040

The load-displacement equations which were set up at Blocks (F) and (G) are solved at this stage in the program. The sequence of operations which reduces the variables by one is identical whether we are concerned with the 1st or the *n*th variable. But each time the variables are reduced in number the region of array E( ) over which operations take place is diminished by one row and one column—compare the sets of equations (4.5) and (4.6). If there is a total of H rows and columns in array E( ), where H = S∗(B + 2), then the reduction operation must be carried out H times. This is the reason why the reduction operation is embraced by the FOR and NEXT statements at lines 1670 and 1980. And whilst N acts as a counter which controls the number of operations, it also controls the region over which these operations are executed. See, for example, lines 1690 and 1740 where, as N increases, the number of rows and columns concerned decreases. Thus, having defined a region, the three following operations are executed within it:

1. To establish the location of the row which has the numerically largest first element (Statements 1680 to 1730);
2. To exchange the contents of this row with the first one in the region—and vice versa (Statements 1740 to 1810);
3. To reduce the number of variables by one (Statements 1820 to 1970).

The first of these operations compares in turn the contents of the first element in each row (E(I, N)) with the contents of K, which was initially set to zero at

```
INPUT BEAM I-VALUES - (MM↑4)
 ? 20000000
 ? 18000000
 ? 36000000
 ? 30000000

INPUT COLUMN I-VALUES - (MM↑4)
 ? 12000000
 ? 15000000
 ? 12000000
 ? 12000000
 ? 15000000
 ? 12000000

INPUT YOUNG'S MODULUS - (N/MM↑2)
 ? 200000

INPUT BEAM UDL'S - (KN/M)
 ? 7
 ? 7
 ? 10
 ? 10

INPUT WIND LOADS - (KN)
 ? 1.5
 ? 3.5


MEMBER END MOMENTS (KNM)
(MEMBERS INTERSECT AT JOINTS DEFINED
 BY THEIR I,J LOCATIONS.)

        I  J        NM                SM              EM              WM
JOINT 1  1          0                 26.1938        -26.1938         0
JOINT 1  2          0                -19.553         -26.4353         45.9883
JOINT 1  3          0                -9.871           0               9.871
JOINT 2  1          24.7              10.0628        -34.7628         0
JOINT 2  2         -18.5742          -13.0075        -35.6502         67.232
JOINT 2  3         -9.19554          -7.35552         0               16.5511
JOINT 3  1          1.9392            0               0               0
JOINT 3  2         -10.369            0               0               0
JOINT 3  3         -6.76996           0               0               0

DISPLACEMENTS

JOINT ROTATIONS (RADIANS)
        I  J
JOINT 1  1         9.93845E-3
JOINT 1  2        -2.92789E-3
JOINT 1  3        -1.21318E-3
JOINT 2  1         8.63133E-3
JOINT 2  2        -2.24272E-3
JOINT 2  3        -6.2215E-4
JOINT 3  1         0
JOINT 3  2         0
JOINT 3  3         0

HORIZONTAL DISPLACEMENTS

                                    DISPLACEMENT
     ROOF LEVEL                    18.9947 MM
   FLOOR 1 LEVEL                   11.1706 MM
FOUNDATION LEVEL                   0 MM
```

```
IF FURTHER LOAD CASES FOR THIS FRAME TYPE 1 ELSE 0 ? 0


IF FURTHER FRAMES TO BE ANALYSED TYPE 1 ELSE 0 ? 0


RUNNING TIME:    6.5 SECS   I/O TIME :   23.1 SECS
```

السيد / الاستاذ الدكتورمحمد حمدى الحفنى الشيخ .

الاسم : الاستاذ الدكتور / محمد حمدى الحفنى الشيخ .

تاريخ الميلاد : ٢٦/٤/١٩٣٧ .

الشهادات الحاصل عليها :
* دكتوراة الهندسة الكهربائية ١٩٦٣ .
* بكالوريس الهندسة الكهربائية ١٩٥٨ .

الوظيفة الحالية : استاد ورئيس قسم هندسة النظم والحاسبات ـ هندسة الازهر .

الخبرة السابقة :
* مشروع السد العالى .
* مشروع المياه الجوفية بالصحراء الغربية .
* شبكات القوى الكهربية (وزارة الكهرباء) .
* شبكات المياة (وزارة الاسكان) .
* دراسات بنوك معلومات مياه النيل (اكاديمية البحث العلمى)
* رئيس قسم الهندسة الكهربية (هندسة الازهر) .

تليفون : منزل ٨٣٧١٠٤ عمل : ٨٣٣٧٢٢

السكن : ٦ شارع يوسف عباس ـ مدينة نصر .