From Professor R.A. Brooker

~~U2782~~
0 0025

University of Essex

Computing Centre
Wivenhoe Park
Colchester CO4 3SQ

Tel: Colchester 44144 (STD Code 020 6)
Telegraphic address: University Colchester
Telex: 98440 (UNILIB COLCHSTR)

RAB/JMW

19th August, 1976

Professor D.E. Knuth,
Computer Science Department,
Stanford University,
Stanford, California 94305.

1st edition is U2780
2nd edition U2781

Dear Donald,

I enclose a photo copy of chapter 3 of the 3rd edition of the Programmers' Handbook For The Manchester Electronic Computer Mart I (September 1953). I think you will find that this explains the techniques of routine organisation developed at that time for a machine with a very small working store. Indeed, you can appreciate that the adroutine virtually invented itself.

If you think it would be useful I could loan you or send you a photo copy of the whole manual but I only have the one copy which I am hanging on to for historical reasons.

With best wishes,

Yours sincerely,

Tony Brooker

R.A. Brooker

Chapter 3

Programming

The coding of a complete problem, a programme, is normally achieved by working up from relatively simple requirements to more complex ones. An example will be used to illustrate the ideas involved, namely, the calculation of the period of a simple pendulum swinging through a finite angle.

Let a be the length of the radius vector from the centre of the circle to the particle and $\alpha = 2\pi R$ the maximum angle which it makes with the downward vertical. It can be shown (H, Lamb, Dynamics, p.107, 2nd edition, Camb. Univ. Press, 1942) that the period of the motion is

$$4\sqrt{\frac{a}{g}} \int_0^{\frac{\pi}{2}} \frac{d\phi}{\sqrt{\cos^2\phi + \left(\frac{1+\cos\alpha}{2}\right)\sin^2\phi}} = 4\,\frac{a}{g}\,K.$$

The ratio which this bears to the period of an infinitely small arc is thus $2\,K/\pi$.

The programme is arranged to read values of R from the input tape, calculate and print the corresponding values of $2K/\pi$. The complete elliptic integral K is computed by a repetitive method based on the Gaussian form of Landen's transformation (Whittaker and Watson, Modern Analysis, p. 533, 4th edition, Camb. Univer. Press, 1950),

$$\int_0^{\frac{\pi}{2}} (a_2^2\cos^2\phi + b_2^2\sin^2\phi)^{-\frac{1}{2}}\,d\phi = \int_0^{\frac{\pi}{2}} (a_1^2\cos^2\phi + b_1^2\sin^2\phi)^{-\frac{1}{2}}\,d\phi$$

where $a_2 = \frac{1}{2}(a_1 + b_1)$, $b_2 = \sqrt{a_1 b_1}$.

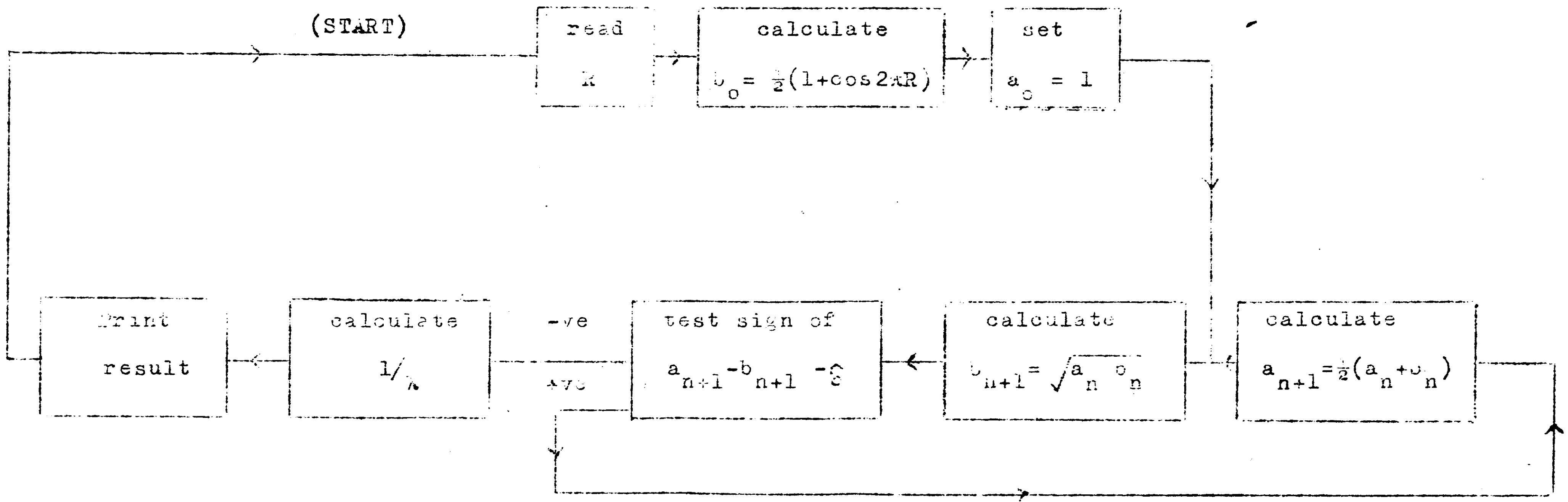
Thus if $a_{n+1} = \frac{1}{2}(a_n + b_n)$ and $b_{n+1} = \sqrt{a_n b_n}$ and λ is the common limit of a_n and b_n, then

$$\int_0^{\frac{\pi}{2}} (a_1^2\cos^2\phi + b_1^2\sin^2\phi)^{-\frac{1}{2}}\,d\phi = \pi/2\lambda.$$

The required ratio of the periods is thus simply the reciprocal of the limit of the sequence with initial values

$$a_1 = 1 \text{ and } b_1 = \sqrt{\frac{1+\cos\alpha}{2}}$$

One way of arranging the calculation is illustrated by the following <u>flow diagram.</u>



(The value of δ, a discriminatory constant, depends on the accuracy required.)

fig. 3.1.

Some of the steps represented are trivial needing only 2 or 3 instructions which can be written down at once in the main programme. For others one constructs a <u>routine</u>, a self contained sequence of instructions and constants for carrying out the process in question. A routine may extend to one or two pages of material. Arrangements are made for calling it in at the appropriate point or points in the calculation.

Using the above flow diagram as a basis one would start to make routines for

1, calculating cos $2\pi x$, given x

2, caloulation of $\sqrt{x}$

3, reading a number punched on the input tape in decimal characters,

4, printing a number in decimal form.

Several questions immediately arise when designing such routines. For example: in which storage location will the <u>cosine</u> routine find the argument x and where will it place the result? What scale factor should be associated with the result.

What is normally required of a routine of this type is that a certain function of the content of one or more lines shall be calculated and stored in a given place, the rest of the content of the store being unaffected by the process, except certain named lines (which should be kept as few as possible) called the working space of the routine. Thus, e.g., the specification of the cosine routine might read as follows

Name: COSINE

Storage Space Occupied: Tube 1.

Effects: Replaces L by $\frac{1}{2}\cos 2\pi[L]_{\pm f}$.

Lines Altered: A, D, and B7, but no store lines outside tube 1.

<u>Library routines</u>

It is clear that certain processes will be common to almost all jobs, e.g., reading, printing, and division. Other processes for which it would be useful to have ready made routines are the following.

Functions: division, square root, cube root, nth root, trigonometrical, exponential, and logarithm.

Processes: linear algebra, integration.

If a library routine can be used then one is saved the task of coding an original routine and the subsequent business of eliminating errors. Moreover one can be sure that it is reasonably economical both in time of operation and storage space. If library routines are seldom used it is because, being built to meet general requirements, they are efficient for none.

The Master Routine

Consider the example already given. When the routines for the basic processes have been coded, or selected from the library, then it is necessary to arrange that they are called in in the correct sequence and that any intermediate bits of arithmetic or shunting operations that may be necessary are carried out.
This is the task of the master routine. The master routine is then a routine for calculating 1/λ and is said to use the basic routines as sub-routines. The master routine may itself be a sub-routine in a larger problem. And so on. Likewise the sub-routines of any routine may themselves have sub-routines. Eventually one arrives at a routine without sub-routines. These notions can be defined more precisely as follows. A routine A is said to use a routine B as a sub-routine if at some point the sequence of instructions of A are interrupted in order to carry out the instructions of B.

In order to return to routine A when B has completed its task it is necessary to preserve a record of the point of departure from A. This record is known as the link. If B itself uses a further sub-routine C then two links have to be preserved. And so on. At any stage in the programme the list of links forms a chain of control extending back to the master routine A. The link from B to A is said to occupy the first level; that from C to B, the second level; and so on. It is very seldom that more than two or three levels are involved.

Storage of routines

In the following conventional scheme for organising a complete programme the routines are kept on a half track or full track of the magnetic store, and transferred to the working (electronic) store when used. With each routine is associated a cue, a 40 digit line pair made up as follows.

Digits 0-9 give the control number, i.e., one less than the name of the line in which the routine starts.

Digits 10-19 are irrelevant.

Digits 20-39 give the magnetic half cue, the magnetic instruction which transfers the routine to the electronic store.

Example: S@//E@/@ is the cue of a routine which is transferred from the left half of track 65 to tube 1 and which starts with the instruction in line I@.

A page of material may contain more than one routine. In such cases the cues of these routines have the magnetic half cue in common.

When calling in a sub-routine the main routine, A, must give the cue of the new routine, B. The link, already defined, is the cue for the return from B to A.

The following outline gives the simplest form which the conventional scheme takes. Several refinements are possible in the interests of time economy but these will be described later.

1. All the routines are stored in the magnetic store on half tracks or full tracks according to their size.

2. Each routine is transferred to the electronic store when required; to tube 0 or tube 1, or tubes 0 and 1 in the case of two page routines.

3. To replace one routine by another and to manage the list of links is the purpose of a special sequence of instructions standing in lines JS to WS inclusive of tube 2. This is the routine changing sequence (R.C.S.). Each new routine is entered through this sequence and control returns to this sequence when the routine has completed its task.

The list of links (the cueue) is kept in line pairs HS, FS, etc.

4. The rest of tube 2 is occupied by the (now familiar) powers of 2 (lines /: to RS inclusive). All this material is called PERM. It is transferred from the magnetic store to tube 2 at the beginning of a programme and, as its name implies, remains there throughout the programme.

5. The rest of the working store, tubes 3,4,5,6, and 7, is available as numerical working space for the routines.

Calling in routines: the routine changing sequence

Each routine must call in the next which may be a sub-routine on the next level or a main routine on the previous level. A further relationship is possible however and is a consequence of limiting routines to two pages. (Most routines are less than one or two pages if only because they correspond to the "relatively simpler requirements"). Occasionally however, and this applies mainly to master routines, routines do extend to 3 or more pages. In such cases they are broken down into two (or possibly even more) routines, A and B, such that when A has completed its task it calls in B on the same level. In this relationship no link is involved: the new routine simply replaces the old. B is called an ad-routine of A.

Two further terms will now be defined. A closed routine is one which ultimately terminates by returning to a routine on the previous level. An open routine is one which terminates by calling in an ad-routine. The prefixes sub- and ad- describe a relationship between two routines whereas the terms open and closed describe a property of a particular routine. It is possible to use either an open or closed routine as a sub- or ad-routine.

The purpose of the routine changing sequence is to enable routines to be changed without too much preparatory bother in the routine being left. Thus to call in a sub-routine the following instructions are inserted in the main routine.

```
        (M=O)

n-1 | s,  / J | M' = cue of new routine
  n | n,  Q O | copies itself into B7
n+1 | F S / P | enters R.C.S.
n+2 |         |<- re-entry.
```

The line pair s contains the cue of the new routine. When the sub-routine has completed its task the main routine is re-entered at line n+2.

To call in an ad-routine the instructions are

```
        (M=O)

n-1 | s,  / J |] replaces cue of current routine
  n | / / E A |] by cue of new routine
n+1 | t,  / P |  enters R.C.S., where t contains
    |         |  the control number TS.
```

A closed routine is terminated simply by giving the instruction LS/P.

The significance of these instructions is revealed when the mechanism of the routine changing sequence is understood. This is:-

```
entry for    -> | J | @ / E A |   adds new cue to list
sub-routines    | N | M £ £ £ |   = -4
                | F | R S Q G |   forms and plants control
                | C | / / Z Z |   number part of the link
from closed     | K | N S Z G |   +4 }
routines     -> | T | A : Z G |   -2 }    +2 adjusts Bl.
entry for    -> | Z | E / E : |   transfers new routine
ad-routines     | L | K S / G |   dummy stop, control number
                | W | / / E P |   enters new routine
                | H | 1st lev | }
                | Y | -el cue | } CUEUE, or chain of
                | P | 2nd lev | } cues and links
                | Q | -el cue | }
```

The chain of links is kept in line pairs HS, PS, OS, corresponding to the first, second, and third levels, etc. The number of levels involved is recorded in Bl. Initially the (original) cue to the master routine stands in HS and Bl contains HSI/. When a sub-routine is called in the cue of the new routine is placed in PS. From n,QO, held in B7, the control line (n+1),QO is formed and copied into HS thus converting the original cue to a link. Bl is adjusted to read PSI/. Finally the magnetic instruction and the control number of the new cue are used to bring down and enter the new routine. Similarly at higher levels: whenever a sub-routine is called in [Bl] is increased by 2; and subsequently decreased by 2 on exit from a closed routine.

In the case of an ad-routine the link is omitted and [B1] remains unaltered. B1 can only be used by a routine if this parameter is temporarily stored elsewhere and replaced before re-entering the R.C.S.

The /C stop was inserted to facilitate "peeping" (see Chapter 8): it enables one to make a visual examination of the routine transferred before its instructions are obeyed.

Further conventions

Design of closed sub-routines

It is now possible to answer some of the questions raised earlier about the design of sub-routines.

Lines altered by the routine should as far as possible be confined to the tube or tubes occupied by the routine itself. To achieve this use can be made of the lines occupied by instructions - after these have been used - as they will be refreshed when the routine is recalled from the magnetic store. Unless this is necessary however routines should be made self resetting so that they can be used as electronic routines (see below).

If in spite of these devices special working lines in other tubes are needed then use can be made of some of the locations j, j+2, j+4, etc., where j,I/ is the quantity recorded in B1.

All routines will make use of some information handed on by the routine it replaces. Thus function routines require an argument. Where should such parameters be located during the transition from one routine to another? The important cases are dealt with separately.

(a) A single 40 digit line is involved. This should be located in the lesser significant half of the accumulator. Thus if the quantity stands originally in VU (say) then the sub-routine can be entered with the sequence

n-2	V U T /	L' = x
n-1	s, / J	
n	n, C O	
n+1	F S / P	

(b) Two 40 digit lines are involved. The second line can be D. Alternately one may wish to preserve both L and M during transition. Two cases arise.

```
(i)        s,  T /     plants cue directly
           @ / Z A
           set A
  n        n,  Q O
  n+1      K S / P     enters R.C.S. at line FS.
```

(ii) The second case arises where A is set by a previous sub-routine. (There is no difficulty in this because A is not **altered when** returning from a closed routine).

Fewer instructions are needed if the first sub-routine is arranged to dump the second result in (say) J+2 and the second sub-routine to recover it from this dump; or alternatively to use the dump directly as a working line. This suggests the convention that sub-routines should terminate with L as one result, M = 0, and the second result, if any, in the dump given by [B1] + 2, that is by [B1] + 4 in the main routine.

(c) If many lines are **involved then these** should be located entirely within a **single** tube, or tube pair, rather than overlap different tubes.

(d) Twenty digit parameters, e.g., names of lines, counters, can be located in B lines. (See convention below regarding the use of B lines in general.)

Electronic routines

No conventions have yet been given about the use of the tubes, 3, 4,5,6, and 7, beyond saying that they are available as working space for routines. If they are not needed in this role, then they may, as a time economy measure, be used for the (more or less) permanent storage of frequently used routines.

These electronic routines can be used in the usual way simply by inserting a dummy magnetic instruction (..£. see fig 1.2) in place of the magnetic half cue. It is necessary that such routines **shall be self-resetting** and designed for the tube in which they are to be stored.

In this connection it would be helpful if several versions of each library routine existed, one for each tube (other than tube 2).

The electronic routines must of course be transferred from the magnetic store initially. This can be arranged by the master routine. The question of how the master routine and PERM are initially transferred will be answered in the next chapter.

B tube conventions

These are:

(1) At the end of a routine B0 = //// unless otherwise stated. Indeed any alteration of B0 should be mentioned in the account of a routine.

(2) Alterations of B lines other than B7 must also be mentioned.

(3) Where a choice of B lines is available, B7 and the higher numbered B lines are to be preferred.

Cues

Cues are normally located in the routines in which they are used. Alternatively they can be assembled as an ordered list - a directory - in one of the tubes 3,4,5,6 or 7.

Replaceability conventions

Certain features of the machine are regarded as disadvantageous and it is desirable to leave open the possibility that these features might at some time be removed. This suggests the convention that no disadvantageous feature of the machine should be used in a library routine.

The features in question are mainly the function codes which are not defined in the table fig 1.3. By trial one can find out what their effects are but the maintenance staff are not responsible for the reliability of these functions. The exceptional nature of the line pairs at the end of an electronic pair is disadvantageous. Possible modification would consist in bringing these into line with the other line pairs.

The official account of a routine

The official account of a routine is designed to provide the user with all the information he needs to incorporate the routine into his programme. This information is divided in to the following sections. See, e.g., the routines of Chapter 5.

Name of routine Needed for identification purposes in verbal or written discussion. It should give some idea of the purpose of the routine.

Purpose A short description of the object of the routine.

Cue The information needed to form a cue for calling in the routine is given.

Magnetic storage: half track or full track; except for routines stored permanently on isolated tracks (see page 4.1) the word variable is also entered.

Electronic storage: the name of the tube or tubes in which the routine operates.

Control numbers: one less than the name of the line in which the routine starts. (There may be alternative entries).

Self-resetting or non-resetting: cross out which does not apply.

Link Describes how the routine is left, namely, whether it is closed or open, or whether further sub-routines are called in. Details of any ad-routines or sub-routines involved are given.

Input details See next chapter.

Effects

A complete account of the effects of the routine would amount to a set of equations giving the new content of any line altered. This might be tedious to compile but the conventions enable certain items to be omitted. No statement need be made about the new content of lines altered in the tube(s) containing the routine itself, or of B7 which is used by the R.C.S. Otherwise the information is given in a table supplemented by notes.

The Instruction Sheet

In addition to the official account the list of instructions is given. Explanatory notes are included where these are thought to have some educational value. The following conventions are observed when drawing up the list of instructions.

Lines or characters left blank indicate that they are neither relevent to, or altered by, the operation of the routine.

Lines altered are indicated thus: if the initial content of a line is irrelevant, then it is filled with the four characters USED; other lines altered are filled in with the initial content and the names of such lines are noted at the foot of the instruction sheet.

Control transfers are ruled in. Unconditional transfers of control are underlined.

Further conventions are introduced in the next chapter.

An example of a master routine

As an illustration of the conventions introduced so far the master routine for the problem given at the beginning of this chapter will be coded. The flow diagram, fig.3.1. is followed very closely. Routines for the basic processes are selected from the library routines described in Chapter 5. These routines are allocated magnetic storage as follows.

DECINPUT	20 R
DECOUTPUT	21 R
COSINE	22 R
SQUAREROOT	23 R
DIVISION	24 R

(There is no particular significance in the choice of right half tracks for the storage of these routines. Any half tracks, left or right could be used).

SQUAREROOT is used as an electronic routine being transferred to tube 1 by a separate instruction in the MASTER routine.

The question "how is the master routine started?" is postponed for the next chapter.

MASTER ROUTINE

/	Y / T :	
E	D S T /	sets $a_1 = 1$
@	@ I T A	
A	Y E / J	calls in
:	: / Q O	DECINPUT: read R
S	F S / P	
I	W E / A	rejects M, if any
U	B E / J	calls in COSINE:
½	½ / Q O	$L'_f = \frac{1}{2} \cos 2\pi R$
D	F S / P	
R	I S T I	$L'_f = \frac{1}{2}(1+\cos 2\pi R)$
J	W E / A	rejects carry (if any)
W	X E / :	transfers SQUARE ROOT routine to tube 1
F	/ / / P	jump to set 6.
C	I S T /	
K	I S / C	
T	/ I / N	$M'_f = \frac{1}{2}(a_n + b_n)$
Z	@ I / N	$= a_{n+1}$
L	@ I / C	
W	@ I / A	plants a_{n+1}
H	/ I / N	$M'_f = a_n b_n$
Y	W E / U	$L'_f = a_n b_n$
P	" E / J	
Q	Q / Q O	$L'_f = \sqrt{a_n b_n} = b_{n+1}$
O	F S / P	
B	/ I T A	plants b_{n+1}
G	@ I T ½	
"	/ I T N	$a_{n+1} - b_{n+1}$
M	S : T N	$a_{n+1} - b_{n+1} - 2^{-37}$
X	R E / H	tests for convergence
V	/ I T /	setting for finding
£	D S / C	reciprocal (see account of DIVISION).

/	V E / J	calls in DIVISION
E	E E Q O	$M'_{\pm} + L_f = \underline{1}$
@	F S / P	
A	V E / A	plants cue for
:	Q E / J	DECOUTPUT: a further
S	@ / E A	alternative to those given
I	W E / J	on page 3.16.
U	N E P O	setting for 12 digits
½	½ E Q O	calls in DECOUTPUT
D	K S / P	to print or punch
R	F / / L	stop
J	D S / P	return to read new value of
N	J / / /	
F		
C		
K		
T		
Z		
L		
W	U S E D	dustbin
H	U S E D	
Y	£ E	cue to DECINPUT
P	H / E @	
Q	£ E	cue to DECOUTPUT
O	Y E E @	
B	S @	cue to COSINE
G	P / E @	
"	£ E	(electronic) cue for
M	£	SQUAREROOT
X	Q / E @	mag-instr, for transferring SQUAREROOT
V	£ E	
£	O / E @	cue to DIVISION

Strategical considerations

Conventions

The whole of what has been said about programming is an elaborate convention regarding the use of the machine and the user can ignore it altogether if he considers it desirable.

The conventions should not be regarded as absolute commands or prohibitions but rather as normal procedure any deviation from which must be noted in the descriptions of programmes and routines. Indeed they enable one to reduce considerably the lengths of official accounts since they allow a great deal to be taken for granted.

The basic plan

If the problem is a genuine numerical computation one must decide what mathematical formulae are to be used. For example if one were calculating the Bessel function $J_0(x)$ one would have, amongst others, the alternatives of using the power series in x, various other power series with other origins, interpolation from a table, various definite integrals, integration of the differential equation by small arcs, and asymptotic formulae. It may be necessary to give some consideration to a number of these taking into account the supply and demand of the economic factors involved. Thus a balance must be struck between the following incompatible desires.

To carry through the process as fast as possible on the machine.

To use as little storage space as possible.

To finish the programming as quickly as possible.

The state of the market for these economic factors will vary greatly from problem to problem. For instance there will be an enormous proportion of problems where there is no question of using the whole storage capacity of the machine, so that space is almost free. With other types of problem one can easily use ten million digits of storage and still not be satisfied. The space shortage applies mainly to working space rather than to space occupied by the routines. Since these usually have to be written down by some one this in itself has a limiting effect. Speed will usually be a factor worth considering, although there are many 'fiddling' jobs where it is almost irrelevant. For instance the calculation of tabular values for functions which are to be stored in the machine and later used for interpolation, would be in this class.

Programmers' time will usually be the main factor in special jobs, but it is relatively unimportant in fundamental (library) routines which are used in most jobs. Accuracy may compete with machine time, e.g., over such questions as the number of terms to be taken in a series, and with space over the question as to whether 20 or 40 digits of a number should be stored.

The available storage space must be allocated to various duties. This applied both to magnetic and electronic storage. It should be possible to estimate the space occupied by instructions to within (say) two tracks, for a large part will probably be previously constructed programme occupying a known number of tracks. The quantities to be held in the working space should, if possible, be arranged in packets which it is convenient to use at once, and which can be packed into a track or half track with the minimum waste of space. For instance when multiplying matrices it may be convenient to partition the matrices into four rowed or eight rowed square matrices and keep either two of the former kind in a half track or each of the latter kind in a full track. The duties of the electronic store are partly ruled by the conventions already introduced but there is still a great deal of freedom, e.g., tubes 3, 4, 5, 6, and 7 are available for working space or electronic routines.

If questions of time are at all critical the plan should include a little detailed coding of the inner loops because these are likely to consume most of the time.

The beginner will do well to ask for advice concerning plans. Bad plans lead to programmes being thrown away.

Coding the individual routines.

As with programming a whole problem a plan is needed for a routine. A convenient aid in this is a detailed flow diagram. The operations appearing as blocks may soon be replaced by actual instructions. It is not usually worth while at first to write down more than the last two characters of the presumptive instruction, i.e., the B-line and function parts.

These are quite enough to remind one of what was the purpose of the instruction. One may then write the instructions into a page deciding at the same time what are to be the addresses involved. Some of the finer points of this have been described in the last chapter.

Manoeuvring space It is seldom that one writes down a page of instructions for the first time without having forgotten one or two vital instructions. It is therefore considered desirable to aim not at pages which are chock-full, but ones which are (say) five-sixths full. The extra space is best left between sequences of consecutive instructions so that one sequence may be extended without interfering with another.

Alternative entries It is often necessary to have a number of routines differing in certain minor particulars. One convenient method is to use one assembly of instructions with different points of entry. The cues of these routines will then differ in their first ten digits. The routine DECOUTPUT provides an example of this. There is one alternative entry by which the carriage return and line feed operations are omitted, causing the number to be printed on the current line of the page.

Space economy measures

Some space economy measures were described in the last chapter. Their use may be justified for time economy reasons. By compressing the instructions into two pages means that they can be called in by a single magnetic transfer. Otherwise they must be arranged as two ad-routines involving extra magnetic transfers and consequent loss of time. Furthermore the fewer the instructions then the better chance there is of being able to use them as an electronic routine.