The construction of examination timetables by computer

By E. Foxley* and K. Lockyer†

In a University which was establishing new departments, increasing student numbers and widening the availability of cross-faculty courses and optional studies, the preparation of examination timetables by manual methods was proving a long and arduous task which rarely produced a timetable even approaching an ideal solution. During 1965 an attempt was made at Nottingham University to see if a successful answer to the examination timetable problem could be found by using the computer. A practicable program was achieved and has now been operational for two years.

(Received March 1968)

The Nottingham program was developed from the method described by Dr. A. J. Cole (Cole, 1964), and familiarity with the method used by Cole will be assumed in this paper. In particular we use Cole's notation and define an examination to consist of a group of one or more papers, all satisfying identical conditions. If two papers differ in any way in their requirements, they must be considered as separate examinations.

The Nottingham program uses the following essential features of Cole's method:

(A) The incompatibility table, being a square matrix of bits indicating which examinations must not occur simultaneously.

(B) The ordering of examinations by some priority criterion so that those with the highest priority are the first to be considered.

(C) A condition for ensuring that certain examination papers should be separated by at least one period.

The main differences between the new implementation and Cole's original method have been made possible by the availability of a larger computer (an English Electric KDF9 compared with the Elliot 803). In particular, the priority criterion has been made more complex and the philosophy concerning the separation of examination papers by free periods has been revised. These differences will now be described in detail.

Priority criterion

In Cole's program the priority of each exam paper is determined by the number of clashes. The more papers with which a given paper clashes, the higher its priority. If two or more papers have an equal number of clashes, priority is given to the exam paper with the highest Succession Coefficient. The Succession Coefficient is 2 if the paper concerned is one of a group which must not occur in adjacent periods, 1 if belonging to a group which must occur in successive periods, and 0 otherwise. If two or more exam papers still have equal priority the number of papers remaining to be allocated is used to determine priority. The exam papers are ordered according to priority level and this ordering is not changed throughout the course of the solution.

In the Nottingham program the priority criterion is more general. The priorities are generated by a subroutine, which can use any formula involving all known facts concerning the examinations. The character of the resulting timetable can be changed by changing the emphasis of the priority formula. Also, unlike Cole's program, in the new implementation examinations can be re-ordered at any time if a new priority ordering is felt to be propitious.

Even with this sophistication, it was felt necessary to have some manual control over an examination's priority. For example an early problem was the tendency of the computer to produce a tail of one or two exam papers per period for several periods after the main bulk of examinations had been allocated. To overcome this trouble each examination is now allocated a priority number between 0 and 15. This priority is set by hand in the 'special data' section of the input and overrides all other priority criteria. This enables examinations which are known to be difficult to fit in, or which are required early, to be given special priority. For example at Nottingham we give high priority to all final year papers, so as to ensure that they will be allocated early in the timetable, thus allowing time for scripts to be sent to the external examiners.

Mention should be made at this point of the criteria by which a timetable is to be judged satisfactory or not. All the work on this project has been shared by the Computing Centre and Examinations Department, and timetables produced by computer have been judged by persons experienced in the allocation of examinations. By varying the subroutine used for generating the priorities, timetables can be produced to minimise the overall examination period, or to allocate the largest examinations (which produce heavy marking loads) early, or to spread each student's load as much as possible. These aims are incompatible. On multiple test runs, the timetable considered most satisfactory by

* *Cripps Computing Centre*
† *Examinations Department* } *University of Nottingham.*

the Examinations Department was about a day longer than the shortest, but was far superior to the shortest in other criteria. Needless to say, it was superior in all criteria to hand-produced timetables.

Separation of examination papers

In order to spread the examination load Cole's program allows for the papers of a given examination to be allocated so as not to appear in adjacent periods. The important thing, however, is not to spread out papers examining a particular subject but to separate the papers taken by any given student. The Nottingham program changes the emphasis from subject to student, so that one can specify one or more free periods before an examination paper for all students taking it, for example we specify a free period before all Finals Papers. This is done by associating with each examination a 'waiting periods' counter. When it is possible to allocate the examination without causing incompatibilities this counter is set to the number W of free periods required. The examination's incompatibilities are then allocated to the next W periods, but room accommodation only to the last of these periods. All 'waiting periods' counters are cleared at weekends, since a weekend is considered to provide sufficient free time for any student to recover.

Clerical preparation

Each student completes an examination entry form on which he details the title and reference number of all the examinations he will be taking. These forms are scrupulously checked for errors and omissions, and used as the basis for the input data. The Examinations Department then produces an overall examination summary which is used to prepare mark sheets and attendance registers, and assists in the abbreviation of the input data (see Section 4 below) if several students are seen to be taking an identical set of examinations.

Program sequence (and sequence of data input)

(1) Read in initial data determining amount of store required. This involves total number of examinations, rooms available, room sizes, etc.

(2) Read in list of examinations to be ignored. These are given a special code number which causes them to be ignored in later calculations. Such examinations include Practicals and others taken outside the normal examination period.

(3) Read in examination coincidences; sets of examinations which must coincide. Coincidence is achieved by treating each set as one examination for allocation purposes. The first set is given the code no. 1 and the second 2 and so on.

(4) Read in one line of data for each student detailing the examinations to be taken, as in Cole's implementation, but with the refinement that if several students are taking exactly the same combination of examinations one single line may be entered, terminated by the number of students taking that set. As each line is read, each examination not previously encountered is given a code number (numbering is consecutive following the numbers already given the coincident papers under paragraph 3) and a vector is set up with bits, one for each code number involved. This vector is 'or'ed into the incompatibility matrix as detailed by Cole. Alongside this operation the computer builds up a running total of students involved in each examination.

When all this data has been read a summary of the incompatibility matrix is printed out. This summary details the examination (or the examinations where a set is involved); the code allocated to it; the number of students taking each paper (or papers if a set) and a list in numerical order of the examinations with which it is incompatible. This incompatibility table has proved invaluable when manipulation to suit special conditions proves necessary. (See **Fig. 1.**)

(5) Read in length of week in periods and the period number in which the examinations are to start, together with the standard examination conditions. Possible conditions which can be specified are given in the next paragraph, and conditions specified at this point apply to all examinations unless others are requested in the next section.

(6) Read in details of any special examination papers not satisfying the standard criteria. The following conditions can be specified:

(*a*) The examination may consist of several papers (a maximum of 7 in our implementation).
(*b*) The examination papers must be in successive periods.
(*c*) The examination papers must be preceded by up to 7 free periods. If condition (*b*) holds the first paper must be preceded by this number of free periods; the rest will be successive. Free periods are not allocated if the exams start on a Monday morning.
(*d*) The examination papers must be allocated to a morning period. A Saturday morning is not counted as a morning period. If condition (*b*) is requested the first paper will be in a morning period.
(*e*) The examination must be allocated to a particular room.
(*f*) The examination paper must be held in separate special accommodation.
(*g*) The examinations are to have specially manually allocated priority.

This special information is inserted in the 'information words' associated with each examination. In our implementation the total of 96 bits is used to store information as shown in **Table 1.**

For a set of coincident examinations the information words must be added so that only one is stored. The information words are in fact associated with an examination code rather than with the examination itself.

CODE NO 34 EXAM NO 886 887
I.C. WITH 634 636 638 645 648 886 887 888 889 890 891
924 947 951 952 953
CLASHES 18 CANDTS 12

CODE NO 35 EXAM NO 888 889 892
I.C. WITH 362 634 636 638 645 648 886 887 888 889 890
893 908 909 924 947 951 952 953
CLASHES 21 CANDTS 15

CODE NO 36 EXAM NO 893 924
I.C. WITH 362 524 525 528 529 530 532 533 535 536 537
634 636 638 645 648 694 811 819 886 887 888 889
892 893 903 904 906 907 908 909 922 923 924 925
952 953
CLASHES 43 CANDTS 50

CODE NO 37 EXAM NO 901 910 911
I.C. WITH 315 317 362 455 456 457 458 493 494 524 525
528 529 530 531 532 533 534 535 536 537 538 539
819 821 829 841 842 850 851 852 853 854 855 856
861 862 896 901 902 910 911 945 946 947 951 952
970
CLASHES 56 CANDTS 63

CODE NO 38 EXAM NO 947 951 952 953
I.C. WITH 361 362 363 364 365 366 528 529 532 535 537
634 636 638 645 647 648 694 814 815 816 817 818
822 823 824 825 826 827 828 829 830 831 832 845
848 849 850 851 852 853 854 855 856 857 858 867
871 872 873 874 875 876 877 878 879 881 882 883
886 887 888 889 891 892 893 901 908 909 910 911
924 925 926 927 928 929 930 931 932 947 948 951
952 953 961 962 963 964 965 966 967 968 969 970
CLASHES 109 CANDTS 291

CODE NO 39 EXAM NO 849 961
I.C. WITH 361 365 366 814 815 816 818 845 846 847 849
951 952 953 961 962 963 964
CLASHES 20 CANDTS 62

CODE NO 40 EXAM NO 874 966
I.C. WITH 362 363 364 854 867 868 870 871 872 873 874
877 879 947 951 952 953 965 966 967 968 969 970
CLASHES 25 CANDTS 32

Fig. 1

(7) Read in all details of precedences in the form of pairs (*a*, *b*) to indicate that all papers of the examination (*a*) must precede all papers of examination (*b*). This information is set up as a two-column matrix with entries (*a*) in the first column and (*b*) in the second.

Allocation of timetable

The computer now prepares the timetable one period at a time. Having ordered the examinations according to the chosen priority criterion the first examination paper is examined to see whether it can be fitted in. If so its row from the incompatibility matrix is 'or'ed into the incompatibility vector for that period. Its students are then allocated to the smallest sized room which can accommodate them, 1 is subtracted from the 'Papers to go' counter, and the allocated flag is set.

For each period the total set of examinations is scanned three times and allocation is done in the three Phases as follows:

Phase I

All examinations whose allocated flag is set are examined. These are examinations which were set in the previous period. If the examination is one whose papers must be in successive periods the examination is reallocated to this period, in all other cases the allocated flag is removed.

Phase II

All examinations accepted during a previous Phase III that are waiting compulsory free periods have their incompatibilities allocated and their waiting periods counter decreased by one. They are allocated accommodation (and their allocated flag is set) only if space is available and if they have waited the appropriate number of periods.

Phase III

The remaining examinations are considered in order, to see if they can be allocated. If it happens that a selected paper must be preceded by one or more free periods its incompatibilities are allocated but its accommodation is not, and its 'waiting periods' counter is set to the appropriate value. It will then be allocated in a later Phase II.

As each examination paper is allocated, the allocation flag is set in its information word.

Table 1

Layout of information word

No. of bits	*Name*	*Use*
16	Code number	Set when examination is first encountered on input
16	No. of students	Accumulated on input. Set to zero if condition (*f*) is required
16	No. of incompatible exams.	Calculated during output of incompatibility matrix. This gives an indication of how difficult it will be to fit the examination in
9	Room no.	Set for special condition if required, otherwise set when the examination is allocated
8	Period to which examination has been allocated	Set on allocation
3	Total number of papers	
3	Number of papers remaining	
3	Number of free periods required before examination	
3	Periods waiting	
1	Exam. allocated flag	Set when the examination is allocated
1	Morning papers flag	Set if this condition is required
1	Successive papers flag	Set if this condition is required
12	Computed priority	Calculated by a subroutine
4	Manual priority	Inserted within special conditions

Printout

For each period the computer prints out the following information:

(*a*) Examination papers allocated to the period.
(*b*) The room to which each paper is allocated.
(*c*) The number of candidates taking each examination.

It also prints a summary after each period detailing the number of seats used and vacant in each room, and a list of all the examinations papers which could have been allocated to the period but for lack of accommodation or the fact that they had been allocated previously. This information is very valuable if any manual manipulation is necessary. (See in **Fig. 2.**)

PERIOD 6 THURSDAY MORNING

ROOM	CANDTS	EXAM	
1	5	177	
1	5	179	
10	69	542	
6	40	503	
11	73	544	
2	14	574	
3	10	641	
1	4	577	
11	227	945	
4	14	499	
5	22	213	
4	11	341	
7	35	404	423
8	40	693	
5	12	908	
7	12	881	885
3	10	137	
9	36	46	
12	38	700	705
2	5	290	
11	22	376	
9	11	941	
12	17	930	
12	15	80	
4	4	226	
12	28	561	
12	13	121	
12	15	273	
12	27	36	
8	8	247	
12	16	983	
10	5	72	
5	4	225	
7	2	42	
8	2	230	
9	2	252	

ROOM	SPARE	USED
1	1	14
2	1	19
3	0	20
4	1	29
5	0	38
6	0	40
7	1	49
8	0	50
9	1	49
10	1	74
11	3	322
12	1	169

TO BE ALLOCATED 338

FREE FOR EXAMS

73	90	143	163	170	171
172	173	174	175	224	229
262	322	324	353	354	355
357	258	360	421	451	452
453	454	464	465	466	483
484	485	711	713	894	895
995					

COMPUTING TIME SO FAR 180 SECONDS.

Fig. 2

Following the detailed timetable period by period, the computer prints out a five-column summary: examination paper number, computer code number, number of candidates, period to which the paper is allocated and the room in which the paper is to be sat. Additionally, those examination papers which are one of a set of coincident papers are starred. (See **Fig. 3.**)

TOTAL NUMBER OF PERIODS 37

EXAM NO	CODE NO	CANDTS		PERIOD	ROOM
1	43	22		1	6
2	44	22		3	6
3	45	6		13	11
4	46	6		11	10
5	51	3		11	6
6	52	3		13	11
7	49	5		7	11
8	50	8		5	11
9	47	10		9	3
10	48	10		7	1
11	132	9		3	2
12	138	6		1	3
13	143	3		13	11
14	146	4		7	4
15	144	2		15	4
16	133	8		9	8
17	134	9		5	11
18	135	8		7	10
19	136	1		11	1
21	139	7		13	8
22	142	3		15	4
23	141	2		11	4
24	137	2		1	10
25	140	1		3	2
26	145	4		11	2
27	147	2		9	3
28	148	4		5	4
31	53	47		1	9
32	54	47		13	11
33	55	47		3	11
34	56	47		11	11
35	57	47		15	11
36	58	27		6	12
37	59	24		8	5
38	65	3		8	1
39	77	12		1	1
40	78	12		3	3
41	149	36		4	5
42	155	2		6	7
43	156	2		4	2
44	157	2		8	4
45	150	35		12	10
46	151	36		6	9
47	152	40		2	6
48	154	11		8	12
49	153	4		10	3
50	170	2		8	1
51	60	62		24	11
52	84	19		7	7
53	62	32		8	7
54	61	11		7	7
61	70	57		26	10
62	128	7		3	9
63	71	55		21	10
64	1	59	*	22	10
65	1	59	*	22	10

Fig. 3

Results

After a successful experimental run in 1965, the computer was used to produce timetables for 1966, 1967 and 1968 summer examinations. The timetables produced were in both cases excellent and required only the minimum of manual adjustment which in almost all cases resulted from requests and restrictions notified after the timetable had been prepared. These adjustments were in any case easy to achieve using the incompatibility table prepared by the computer. The most pleasing achievements have been the reduction in the overall length of the examination period from the thirty sessions achieved manually in 1965 to only twenty-three periods in 1967 (well over half of the examination papers were allocated into the first eleven periods), the remarkably equitable spread of examination papers for all candidates, and the excellent utilisation of accommodation.

In 1967 the computer allocated 651 examination papers involving approximately 15,000 candidate-sessions in 23 periods. There were over 800 different combinations of examinations involved. The complete program and data ran in 12K of 48-bit words on the University KDF9, taking six minutes.

Future developments

Future improvements for this program include the computerised printing of attendance registers and mark sheets for examination. To achieve this, a punched card including all relevant information will be prepared for each student from his examination entry form. This development will also eliminate the clerical task of preparing summaries. When this extension has been achieved it is not difficult to see how this program could be built into an integrated system of student records.

Conclusions

Although we do not claim that this program is very sophisticated, it has been proved to produce far better timetables and with far less effort than the previous manual methods.

Reference

COLE, A. J. (1964). The preparation of examination timetables using a small-store computer, *Computer Journal*, Vol. 7, p. 117.