

QUESITI

- 1** Come sono memorizzati i dati e le istruzioni nella macchina di von Neumann?
- A Entrambi in formato numerico.
 - B I dati in formato numerico, le istruzioni in formato alfabetico.
 - C Le istruzioni in formato alfabetico, il formato dei dati cambia in funzione del tipo.
 - D Le istruzioni in formato numerico, il formato dei dati cambia in funzione del tipo.
- 2** Qual è il contenuto del registro PC (*Program Counter*) subito dopo l'esecuzione dell'istruzione «JMP 50»?
- A Non è possibile stabilirlo.
 - B Il valore dell'indirizzo della locazione di memoria che contiene l'istruzione aumentato di 50.
 - C 50.
 - D Dipende dal risultato dell'ultima istruzione eseguita dall'unità ALU.
- 3** La seguente tabella rappresenta lo stato di alcune locazioni della memoria RAM di una macchina di von Neumann:

Indirizzo	Contenuto
100	10
101	0
102	1

Rappresentare lo stato delle locazioni di memoria dopo l'esecuzione delle seguenti istruzioni:

LOD 100
STO 101
LOD #10
STO 102

Indirizzo	Contenuto
100	
101	
102	

- 4** Associare al registro corretto la propria funzionalità.

A	Contiene l'indirizzo della prossima istruzione da eseguire
PC	Contiene il codice operativo dell'istruzione da eseguire
IR	Contiene un operando o il risultato di un'operazione aritmetica

ESERCIZI

- 1** Simulare il funzionamento del seguente programma per la macchina di von Neumann realizzando più tabelle di traccia:

0. LOD 100
1. DIV 101
2. MUL 101
3. STO 102
4. LOD 100
5. SUB 102
6. STO 103
7. HLT

Che cosa calcola il programma nella locazione di indirizzo 103 a partire dai dati contenuti nelle locazioni di indirizzo 100 e 101?

- 2** Simulare il funzionamento del seguente programma per la macchina di von Neumann realizzando più tabelle di traccia:

0. LOD #0	12. ADD 101
1. STO 103	13. STO 103
2. LOD 100	14. LOD 100
3. JMZ 22	15. DIV #2
4. LOD 100	16. STO 100
5. DIV #2	17. LOD 101
6. MUL #2	18. MUL #2
7. STO 99	19. STO 101
8. LOD 100	20. LOD 100
9. SUB 99	21. JMP 3
10. JMZ 14	22. HLT
11. LOD 103	

Che cosa calcola il programma nella locazione di indirizzo 103 a partire dai dati contenuti nelle locazioni di indirizzo 100 e 101?

- 3** Simulare il funzionamento del seguente programma per la macchina di von Neumann realizzando più tabelle di traccia:

0. LOD 100	7. JMZ 13
1. DIV 101	8. LOD 101
2. MUL 101	9. STO 100
3. STO 99	10. LOD 102
4. LOD 100	11. STO 101
5. SUB 99	12. JMP 0
6. STO 102	13. HLT

Che cosa calcola il programma nella locazione di indirizzo 101 a partire dai dati contenuti nelle locazioni di indirizzo 100 e 101?

- 4** Scrivere un programma per la macchina di von Neumann che determini il maggiore tra due numeri X e Y inizialmente contenuti nelle locazioni di memoria di indirizzi 100 e 101, memorizzando il risultato nella locazione di indirizzo 102. Verificarne il corretto funzionamento realizzando più tabelle di traccia.
- 5** Scrivere un programma per la macchina di von Neumann che calcoli il quadrato di un numero N inizialmente contenuto nella locazione di memoria di indirizzo 100, memorizzando il risultato nella locazione di indirizzo 101. Verificarne il corretto funzionamento realizzando più tabelle di traccia.
- 6** Scrivere un programma per la macchina di von Neumann che calcoli il fattoriale⁵ di un numero N inizialmente contenuto nella locazione di memoria di indirizzo 100 memorizzando il risultato nella locazione di indirizzo 101. Verificarne il corretto funzionamento realizzando più tabelle di traccia.

5. Il fattoriale di un numero naturale n (in simboli $n!$) è il prodotto di tutti i numeri naturali compresi tra 1 e n (n incluso); per definizione $0! = 1$.

- 7** Scrivere un programma per la macchina di von Neumann che calcoli la potenza X^Y dei numeri X e Y inizialmente contenuti nelle locazioni di memoria di indirizzi 100 e 101 memorizzando il risultato nella locazione di indirizzo 102. Verificarne il corretto funzionamento realizzando più tabelle di traccia.

- 8** Simulare il funzionamento del seguente programma per la macchina di von Neumann realizzando più tabelle di traccia:

0. LOD 101	16. LOD 102
2. JMZ 24	18. ADD 100
4. LOD 100	20. STO 102
6. STO 102	22. JMP 8
8. LOD 101	24. LOD #0
10. SUB #1	26. STO 102
12. STO 101	28. HLT
14. JMZ 28	

Che cosa calcola il programma nella locazione di memoria di indirizzo 102 a partire dai dati contenuti nelle locazioni di indirizzo 100 e 101?

- 9** Scrivere un programma per la macchina di von Neumann che verifichi se un numero inizialmente presente nella locazione di memoria 100 sia divisibile per uno dei seguenti numeri: 2, 3, 5 o 7. In caso affermativo dovrà inserire il valore 1 («vero») nella locazione di memoria 101, in caso negativo vi inserirà il valore 0 («falso»).
- 10** Scrivere un programma per la macchina di von Neumann che verifichi se un numero inizialmente presente nella locazione di memoria 100 è primo⁶. In caso affermativo dovrà inserire il valore 1 («vero») nella locazione di memoria 101, in caso negativo vi inserirà il valore 0 («falso»).

6. Per definizione un numero N è primo se non è divisibile per nessuno dei numeri compresi tra 2 e $N - 1$ inclusi.

punch-card

scheda perforata

teletype tape

nastro per telescrivente

plug-board

pannello di controllo

faultlessly

privo di guasti

negligible

trascurabile

sound

valido

scrutiny

indagine

purpose

scopo

carry out

eseguire

sum up

riassumere

perform

eseguire

endowed

dotato

1.0 Definitions

1.1 The considerations which follow deal with the structure of a *very high speed automatic digital computing system*, and in particular with its *logical control*. Before going into specific details, some general explanatory remarks regarding these concepts may be appropriate.

1.2 An *automatic computing system* is a device, which can carry out instructions to perform calculations of a considerable order of complexity. The instructions which govern this operation must be given to the device in absolutely exhaustive detail. They include all numerical information which is required to solve the problem under consideration: Initial values of the dependent variables, values of fixed parameters (constants), tables of fixed functions which occur in the statement of the problem. These instructions must be given in some form which the device can sense: Punched into a system of punchcards or on teletype tape, magnetically impressed on steel tape or wire, photographically impressed on motion picture film, wired into one or more fixed or exchangeable plug-boards – this list being by no means necessarily complete. All these procedures require the use of some code to express the logical and the algebraic definition of the problem under consideration, as well as the necessary numerical material.

Once these instructions are given to the device, it must be able to carry them out completely and without any need for further intelligent human intervention. At the end of the required operations the device must record the results again in one of the forms referred to above. The results are numerical data; they are a specified part of the numerical material produced by the device in the process of carrying out the instructions referred to above.

1.3 It is worth noting, however, that the device will in general produce essentially more numerical material (in order to reach the results) than the (final) results mentioned. Thus only a fraction of its numerical output will have to be recorded as indicated in 1.2, the remainder will only circulate in the interior of the device, and never be recorded for human sensing. This point will receive closer consideration subsequently.

1.4 The remarks of 1.2 on the desired automatic functioning of the device must, of course, assume that it functions faultlessly. Malfunctioning of any device has, however, always a finite probability – and for a complicated device and a long sequence of operations it may not be possible to keep this probability negligible. Any error may vitiate the entire output of the device. For the recognition and correction of such malfunctions intelligent human intervention will in general be necessary.

However, it may be possible to avoid even these phenomena to some extent. The device may recognize the most frequent malfunctions automatically, indicate their presence and location by externally visible signs, and then stop. Under certain conditions it might even carry out the necessary correction automatically and continue.

2.0 Main subdivisions of the system

2.1 In analyzing the functioning of the contemplated device, certain classificatory distinctions suggest themselves immediately.

2.2 First: Since the device is primarily a computer, it will have to perform the elementary operations of arithmetic most frequently. These are addition, subtraction, multiplication and division: +; –; ×; ÷. It is therefore reasonable that it should contain specialized organs for just these operations.

It must be observed, however, that while this principle as such is probably sound, the specific way in which it is realized requires close scrutiny. Even the above list of operations: +; –; ×; ÷, is not beyond doubt. [...]. At any rate a *central arithmetical* part of the device will probably have to exist, and this constitutes the *first specific part: CA*.

2.3 Second: The logical control of the device, that is the proper sequencing of its operations can be most efficiently carried out by a central control organ. If the device is to be

elastic, that is as nearly as possible *all purpose*, then a distinction must be made between the specific instructions given for and defining a particular problem, and the general control organs which see to it that these instructions – no matter what they are – are carried out. The former must be stored in some way – in existing devices this is done as indicated in 1.2 – the latter are represented by definite operating parts of the device. By the *central control* we mean this latter function only, and the organs which perform it form the *second specific part*: CC.

2.4 Third: Any device which is to carry out long and complicated sequences of operations (specifically of calculations) must have a considerable memory. [...].

2.5 To sum up the third remark: The device requires a considerable memory. While it appeared that various parts of this memory have to perform functions which differ somewhat in their nature and considerably in their purpose, it is nevertheless tempting to treat the entire memory as one organ, and to have its parts even as interchangeable as possible for the various functions enumerated above. [...]. At any rate the total *memory* constitutes the *third specific part of the device*: M.

2.6 The three specific parts CA, CC (together C) and M correspond to the *associative* neurons in the human nervous system. It remains to discuss the equivalents of the *sensory* or *afferent* and the *motor* or *efferent* neurons. These are the *input* and the *output* organs of the device, and we shall now consider them briefly.

In other words: all transfers of numerical (or other) information between the parts C and M of the device must be effected by the mechanisms contained in these parts. There remains, however, the necessity of getting the original information from outside into the device, and also of getting the final information, the results, from the device into the outside.

By the outside we mean media of the type described in 1.2: here information can be produced more or less directly by human action (typing, punching, photographing light impulses produced by keys of the same type, magnetizing metal tape or wire in some analogous manner, etc.), it can be statically stored, and finally sensed more or less directly by human organs.

The device must be endowed with the ability to maintain the input and output (sensory and motor) contact with some specific medium of this type: That medium will be called the *outside recording medium of the device*: R.

2.7 Fourth: The device must have organs to transfer (numerical or other) information from R into its specific parts, C and M. These organs form its *input*, the *fourth specific part*: I. It will be seen that it is best to make all transfers from R (by I) into M, and never directly into C.

2.8 Fifth: The device must have organs to transfer (presumably only numerical information) from its specific parts C and M into R. These organs form its *output*, the *fifth specific part*: O. It will be seen that it is again best to make all transfers from M (by O) into R, and never directly from C.

[J. Von Neumann, "First Draft of a Report on the EDVAC", 30 June 1945]

QUESTIONS

- a What is an automatic computing system?
- b When is human intervention on the computing device necessary?
- c Where can you store the instructions for the computing device?
- d Which kind of data are the results computed by the device?
- e What are the specific parts of a computing system?
- f Which part of a computing system is the equivalent of the sensory neurons of the human nervous system? And which one is the equivalent of the motor neurons?