

MARKSCHEME

November 2001

COMPUTER SCIENCE

Higher Level

Paper 1

SECTION A

1. 11001010 *[1 mark]*

2. (a) *[1 mark]* for the following, or similar definition:
 - a tree is a hierarchical data structure
 - each child node
 - is below a parent
 - a node which has child nodes below is parent node

- (b) *[1 mark]* for any of the following *[max 2 marks]*:
 - to search for files in a logical order
 - directory as parent node
 - sub directories as child nodes
 - until list addresses of files found

3. *[1 mark]* for any of the following *[max 2 marks]*:
 - carries data, instructions and addresses
 - between CU, ALU and main memory
 - to fetch and execute instructions

[1 mark] for any of the following *[max 2 marks]*. Overall *[max 3 marks]*:

- max processing speed needed
- parallel carries all bits at the same time
- serial would mean one bit at a time so too slow
- immediate access needed

4. *[max of 2 marks]* for advantage and *[2 marks]* for disadvantage *[1 mark]* for valid point and *[1 mark]* for description or justification.

Advantages

- no need to go to the doctor for trivial illnesses which saves time and money
- can be quickly reassured that illness not important
- doctor does not waste time with trivial complaints
- early warning of symptoms that could lead to serious illness
- some people feel too shy to explain their symptoms to a person and feel more secure with a computer.

Disadvantages

- medical expertise not easily transferred to program
- patients may not realise all the symptoms
- many illnesses need personal reassurance
- not a good way to find out that you may have a serious illness
- mistakes in input could have serious consequences in either direction

5. Circular

[1 mark] for any of the following [max 2 marks]:

- confines the list to a predefined area in store
- problems if queue becomes greater than given space
- only two pointers needed but each time item is added have to ensure front and end do not coincide
- and check for wrap around each time an item added or taken
- in the case of wrap around calculation of pointer takes time
- items do not have to be moved

Linear

[1 mark] for any of the following [max 2 marks]:

- if not moved up each time an item taken a lot of storage space is wasted
- very quick to add items as pointers quickly adjusted
- if list moved up when item taken then both pointers have to be adjusted and moving every item in a long list takes time

Allow any valid point on each structure / algorithm to implement the structure.

6. [2 marks] for each feature. [1 mark] for identifying and [1 mark] for explanation:

- rotational delay (latency) disk rotating to appropriate sector
- seek time as heads move to appropriate cylinder
- transfer time to send data from disk to main memory

7. (a) [max 2 marks] with one for each of the following points:

MHz refers to frequency **[1 mark]**

of fetch execute cycles **[1 mark]** per second

in this case 750 mega **[1 mark]** or binary million **[1 mark]** cycles per second

(b) personal computer or workstation or portable **[1 mark]**

8. [1 mark] for:

OS / applications need more memory

[1 mark] for reason why:

use of more complex GUIs, spread of multi-tasking *etc.*

9. *[1 mark] for each valid point up to [max 2 marks]:*

Systems analysis

- system needs to change over time *[1 mark]*
- to incorporate new features *[1 mark]*
- update system in light of how it has performed *[1 mark]*

Code preparation

[1 mark] for each valid point up to [max 2 marks]:

- new sections of code may have to be written *[1 mark]*
- some may need amending in the light of changing circumstances *[1 mark]*
- for example new fields in records *[1 mark]*
- space for more records in a file *[1 mark]*

10. *[1 mark] for each valid point [max 2 marks]:*

- sending computer sends message “ready to send”
- receiving computer sends message “ready to receive”
- handshake established and first computer sends

11. *[1 mark] for each valid point [max 2 marks]:*

- allows one object to be derived from another
- the derived object has all the data members and functions of the original
- plus any extra that are defined within it

12. *[1 mark] for each valid point [max 3 marks]:*

- cost of installing hardware and software for new system
- configuration of possible systems/details of proposed new system
- description of effects of new system on production and workers
- cost benefit analysis

SECTION B

13. (a) Award **[1 mark]** for each correct line and **[1 mark]** for output:

| LEFT | RIGHT | POS | output |
|------|-------|-----|------------|
| 1 | 6 | 3 | |
| 4 | 6 | 5 | |
| 4 | 5 | 4 | item found |

[total 4 marks]

(b) There are various possibilities.

Allocate:

[1 mark] initialise a counting variable `Z <= 1`

[1 mark] note position where found `POS <= POSITION`

[2 marks] for looking to right (*allow [1 mark] for attempt*)

e.g.

```
Z <-- 1
```

```
POS = POSITION
```

```
while VALUE (POS) = VALUE (POS + 1)
```

```
    Z = Z + 1
```

```
    POS = POS + 1
```

```
endwhile
```

[1 mark] for also going left

[1 mark] for terminating with:

```
until LEFT > RIGHT or Z = 0
```

[6 marks]

14.

| a | b | c | lights |
|---|---|---|--------|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

[1 mark] for each 2 rows correct. e.g. 5 rows correct gets **[2 marks]** **[max 4 marks]**

(b) **[2 marks]** for the following expression:

$$(\text{not } a.b.\text{not } c) + (\text{not } a.b.c) + (a.\text{not } b.\text{not } c) + (a.\text{not } b.c)$$

[1 mark] if no more than one term is incorrectly transferred from the truth table.

Allow **[2 marks]** for follow through if truth table is incorrect but expression is correctly derived from the truth table. **[max 2 marks]**

(c) By Karnaugh map

| | c | not c |
|-------------|---|-------|
| not a.not b | | |
| not a.b | 1 | 1 |
| a.b | | |
| a.not b | 1 | 1 |

$$(\text{not } a.b) \text{ OR } (a.\text{not } b)$$

from first and second terms: not a b(c OR not c) = not a b
 from third and fourth terms: a not b(not c OR c) = a not b

Final expression simplifies to a XOR b

[4 marks] for a XOR b; **[3 marks]** for (not a b) OR (a not b).
 Allow follow through **[max 4 marks]**.

15. (a) **[1 mark]** for each of the following **[max 2 marks]** :
- Go to head pointer, compare name,
 - if not equal follow next pointer
 - repeat until name of artist found.
- (b) **[1 mark]** for clear start node.
[2 marks] for clear pointers to next two nodes.
[1 mark] for indicating year and artist pointers.
- (c) **[1 mark]** for each of the following points **[max 4 marks]** :
- stack is used to record the return addresses
 - last one added is first returned
 - by creating linked list pointing to return address each time subroutine called
 - include back pointers
 - and traversing in reverse order to return to correct address.

16. (a) (i) *Accept [1 mark] for each of the following [max 3 marks]:*
- when buffer full an interrupt sent to O/S
 - spell checking halted
 - necessary location addresses put on stack
 - buffer emptied
 - information taken off stack and spell checking continues
- (ii) *Accept [1 mark] for each of the following [max 3 marks]:*
- when buffer full data transferred directly to memory
 - processor not involved
 - spell checking continues unhalting
- (b) address bus *[max 2 marks]*
- address of data needed
 - sent from instruction register
 - opens appropriate path to memory location
- data bus *[max 2 marks]*
- data copied from memory location
 - sent along data bus to accumulator

17. (a) Test data would be generated by some other device **[1 mark]**
and output checked for correct warning signals **[1 mark]**
*Give [1 mark] for making clear that not a real heart used and [1 mark] for output check
[max 2 marks]*

- (b)
- normal data **[1 mark]** that is data within the expected range **[1 mark]**
 - extreme data **[1 mark]** that is beyond normal limits **[1 mark]**
 - abnormal data **[1 mark]** e.g. no signal at all **[1 mark]** too high for a real heart **[1 mark]**
 - data at the limits **[1 mark]** i.e. just inside/outside normal range **[1 mark]**

[max 2 marks] for each type of data. Accept only two answers [max 4 marks].

(c) **[2 marks]** for stating at least 2 different methods of changeover and **[2 marks]** for clearly explained implications.

For example

- parallel running **[1 mark]** so that if a failure in new system the existing one gives backup **[1 mark]**
 - direct changeover **[1 mark]** could be risky with no backup **[1 mark]**
 - phased introduction **[1 mark]** gives staff time to get accustomed to new system **[1 mark]**
-