INTERNATIONAL BACCALAUREATE
BACCALAURÉAT INTERNATIONAL
BACHILLERATO INTERNACIONAL

MARKSCHEME

May 2001

COMPUTER SCIENCE

Higher Level

Paper 1

12 pages

SECTION A

1. *Award **[1 mark]** for correct reason and **[1 mark]** for a correct elaboration:*

e.g. Dirt ***[1 mark]*** would stop OCR, but not MICR ***[1 mark]***.
Fraud ***[1 mark]*** because it is more difficult to forge MICR ***[1 mark]***.

2. 16 × 1024 × 8 = 131072 bits per second ***[2 marks]***.
(*Award both marks if left as 16 × 1024 × 8*)
*Award **[1 mark]** for 16 × 1000 × 8 = 128000 bps.*

If only 16 × 1024 is done, give ***[1 mark]***. No marks for only 16000 bps.

3. *Award **[1 mark]** for each error, and **[1 mark]** for a correct example for each error to give a maximum of **[6 marks].***

Syntax error ***[1 mark]***, any example, *e.g.* misspelling a reserved word *etc.* ***[1 mark]***.

Logic error ***[1 mark]***, any calculation error, use of wrong file *etc.* ***[1 mark]***.

Run-time error ***[1 mark]***, *e.g.* division by zero, no file *etc.* ***[1 mark]***.

4. *Award **[1 mark]** for the following:*

A component that co-ordinates input / output devices.

5. *Award **[1 mark]** for a correct fact about **each**, and **[1 mark]** for a correct example of each, up to **[4 marks]** max.*

Batch processing does not process data immediately, but waits until a group of data has been collected ***[1 mark]*** whereas on-line processing is interactive ***[1 mark]***. *e.g.* of batch processing is payroll / utility billing ***[1 mark]*** an *e.g.* of on-line is air-traffic control / supermarket tills *etc.* ***[1 mark]***.

6. (a) *Award **[2 marks]** max for a description:*

Double-entry of data ***[1 mark]*** both versions compared by software ***[1 mark]*** and differences highlighted ***[1 mark]***.

*Award **[1 mark]** for the error type:*

Any mistyping / misreading of input data by keyboard entry.

(b) *Award **[2 marks]** max for a description and **[1 mark]** for an application.*

Range check ***[1 mark]*** to test if input data is in pre-set limits ***[1 mark]***. Entering percentages in an examination.

Type check ***[1 mark]*** to test if input data is correct data type ***[1 mark]***. Test if percentage is an integer *etc.*

Etc.

7. *Award **[2 marks]** for each correct **direct comparison** up to a max of **[4 marks]**. (If a comparison is attempted, but is not direct, or tenuous, award up to **[2 marks]** max. Award no marks if no comparison is attempted, i.e. just unrelated statements about each).*

Integer values are precise ***[1 mark]*** whereas real values often are approximations ***[1 mark]***.

Integer values have a smaller range than real values ***[1 mark]*** when using the same number of bits ***[1 mark]***.

Integer values require one storage part ***[1 mark]*** whereas real values require two stored components ***[1 mark]*** (the mantissa and exponent).

8. A calculation is carried out on the key field ***[1 mark]*** that generates the record's address / position ***[1 mark]***.

An advantage is that it is quicker ***[1 mark]*** to find a record than other file organisations ***[1 mark]***.

A disadvantage is that data appears unordered ***[1 mark]*** so displaying the file in order is time-consuming / 'difficult' ***[1 mark]*** / clashes often occur ***[1 mark]*** and so overflow has to be dealt with ***[1 mark]***.

9. *Award up to **[2 marks]** for each use, for two applications to give **[4 marks]** max.*

Parameter / return address storage ***[1 mark]*** in the calling of subroutines ***[1 mark]***.

Interrupt handling ***[1 mark]*** to store the return addresses of commands ***[1 mark]***.

Evaluation of arithmetic expressions ***[1 mark]*** to store operator precedence ***[1 mark]***.

10. *Award **[1 mark]** for each feature identified, and a further **[1 mark]** for an elaboration up to **two** features to give **[4 marks]** max. Award **[1 mark]** for any disadvantage.* (*This gives a question maximum of **[5 marks]**.*)

Encapsulation ***[1 mark]*** grouping data and operations ***[1 mark]***.
Polymorphism ***[1 mark]*** the same operation can be applied to different objects, with each object behaving appropriately ***[1 mark]***.
Inheritance ***[1 mark]*** allows one object to be derived from another ***[1 mark]***.

Disadvantages include extra development time (for simple problems) / many languages not supporting objects / lack of standardisation / more difficult to design solutions / need to learn a new programming paradigm (method).

SECTION B

11. (a) *Award marks as follows:*

POS	DATA[POS]=1	COUNT	CHECK	***Marks***
		0		[given]
1	false	0		[given]
2	true	1		[given]
3	true	2		***[1 mark]*** for increment
4	true	3		
5	false	3		***[1 mark]*** for not incrementing
6	true	4		
7	true	5		***[1 mark]*** for final answer of 5
8	false	5	false	***[1 mark]*** for returning false

(b) *Award **[1 mark]** for:*

true.

(c) *Award **[2 marks]** for stating:*

even parity;
If just 'parity' is stated, give ***[1 mark]***.

(d) *Award **[1 mark]** for:*

the idea of communications / transmission / WAN.

(e) *Award **[2 marks]** for a complete answer. e.g.:*

If two bits are changed, then no error is detected.

It can detect an odd number of bits change, but cannot correct it.

It cannot correct the error because the location is not known.

*Award **[1 mark]** for a partial answer. e.g.*
'not all errors are detected', 'cannot correct errors'.

12. (a) *Award **[1 mark]** for:*

9

(b) *Award **[3 marks]** for a clear answer to **each** of the three areas (**[1-2 marks]** for a partial answer) up to a max **[9 marks]**. (Allow **[1 mark]** for a brief, correct, statement, **[2 marks]** for a better, but incomplete answer.)* (NOTE: Do not allow more than ***[3 marks]*** for each section.)

(i) **Hardware**:
Bus width will have to change ***[1 mark]*** to transfer the longer byte between units ***[1 mark]***; Data storage / (main) memory / registers will have to change ***[1 mark]*** to store the larger byte ***[1 mark]***;
Peripherals will have to be adapted ***[1 mark]*** because they will be made for standard byte lengths ***[1 mark]***.

(ii) **Software**:
Programs will have to be adapted ***[1 mark]*** because they will not be made for non-standard protocols ***[1 mark]***.
The drivers for peripherals will have to change ***[1 mark]*** to send the correct signals to the devices ***[1 mark]***.

(iii) **Communication**:
Standard protocols will not use 9-bit bytes ***[1 mark]*** and so will have to change for this data ***[1 mark]***.
It may not be possible to change the protocols ***[1 mark]*** and so communication between systems may not be possible ***[1 mark]***.

13. (a) *Award **[1 mark]** for each part. (Both instruction **and** co-ordinates need to be given for the mark):*

(i) A **line** is drawn from (3,6) to (2,0) ***[1 mark]***.

(ii) A **move** is carried out to (12,45) ***[1 mark]***.
(Do not award the mark if any reference is made to (2,0)).

(b) *Award **[1 mark]** if reference is made to having to store both/two parts, e.g.*

The first array location will store the code of the new shape and 15 co-ordinates require 30 locations (15x and 15y) ***[1 mark]***.

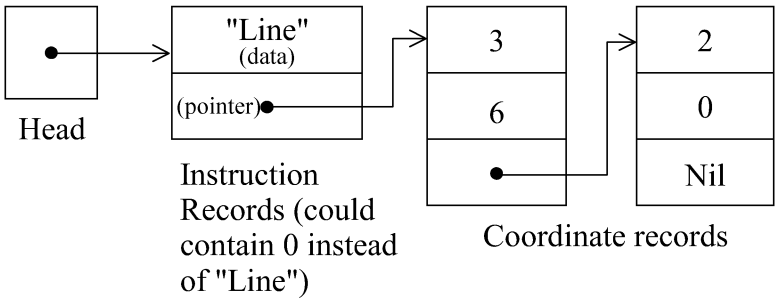
(c) (i) *Award up to **[2 marks]** for a description as to why an array is inefficient (**[1 mark]** for a partial answer) e.g.*

Most of the array locations are not used ***[1 mark]*** and so this is a waste of space ***[1 mark]***.

(ii) *Award **[1 mark]** for a suitable **dynamic** data structure; **[2 marks]** for a clear **labelled** sketch (**[1 mark]** for an incomplete sketch); and **[2 marks]** for an explanation as to why it is better than an array (**[1 mark]** for a partial/vague explanation). This gives a maximum of **[5 marks]**.*

(NOTE: if the suggested data structure is not dynamic, or a file is suggested (which is incorrect, since the question states 'within the program') do not award the first mark, but the other marks are available if well explained and sensible).

e.g.
A linked list (of records) ***[1 mark]***.



(Obviously not so much detail is required, the idea of data/pointers and labelling is required though.) ***[2 marks]***

A dynamic structure is better since only the space required is used, so there is none wasted. ***[2 marks]***

14. (a) *Award **[1 mark]** for any of the points below:*

- test for empty queue;
- test for full queue;
- remove a data item from the **front** (do **not** accept just 'remove the data item');
- display data item at the front.

(b) *Award **[1 mark]** for a correct area of use, and **[1 mark]** for a brief elaboration:*

- printer queue ***[1 mark]*** to store jobs sent in order ***[1 mark]***;
- keyboard queue ***[1 mark]*** to store input from a user (until processor is ready) ***[1 mark]***.

(c) *Award **[2 marks]** for a description of encapsulation, and **[1 mark]** for relating it to the example.*

- the data and the operations that act on it are grouped ***[1 mark]*** into a single 'program unit' called an object ***[1 mark]***;
- *e.g.* the queue structure and the operations add / remove, *etc.* ***[1 mark]***.

(d) *Award marks as follows, up to a maximum of **[4 marks]**:*

The implementation of the queue does not matter ***[1 mark]*** since the programs that use the object ***[1 mark]*** will not have to change ***[1 mark]*** only the coding of the object ***[1 mark]***.

15. (a) *Award **[1 mark]** for identifying a correct factor, up to **[4 marks]** max, e.g.*

- the data required to be input;
- how it is easiest to collect;
- the data required to be output;
- the format the user requires;
- any special considerations;
- such as user handicaps / other tasks (*e.g.* can't use hands because doing something else).

(b) *Award **[1 mark]** for identifying a suitable consideration, up to **[3 marks]** max, e.g.*

- ask user;
- look at current presentation of output;
- offer prototypes to the user;
- view similar packages for ideas;
- design on paper first / sketches;

(c) *Award **[1 mark]** for each valid point, up to **[3 marks]** max, e.g.*

- use current / typical data;
- use extreme / invalid data;
- programmers test for robustness of software;
- (prototype) with user to test;
- feedback / add changes from user.