



22087011



International Baccalaureate®
Baccalauréat International
Bachillerato Internacional

**COMPUTER SCIENCE
HIGHER LEVEL
PAPER 1**

Thursday 22 May 2008 (afternoon)

2 hours 15 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Section A: answer all the questions.
- Section B: answer all the questions.

SECTION A

Answer **all** the questions.

1. State **three** components of a *feasibility report*. [3 marks]

2. Outline **two** reasons for the need for *systems maintenance*. [4 marks]

3. Identify **three** different types of data that should be included as part of a *testing strategy*. [3 marks]

4. Outline **one** difference between the functioning of an *interpreter* and a *compiler*. [2 marks]

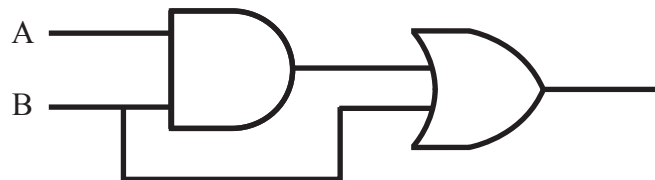
5. Define the term *direct access*. [2 marks]

6. Outline an application that would make use of a *touch screen* for input of data. [2 marks]

7. Evaluate the *hexadecimal* expression $AB + B$, giving your answer in
 - (a) hexadecimal; [1 mark]
 - (b) binary. [1 mark]

8. Define the term *underflow*, when used in connection with binary arithmetic. [2 marks]

9. Write a **simplified** Boolean expression for the following circuit.



[2 marks]

10. Outline the functions of the following *registers*:

(a) instruction register; [2 marks]

(b) accumulator. [2 marks]

11. Consider the following *recursive algorithm*.

```
int recur(int b)
{ if (b==0)
    return 1;
  else
    return 2*recur(b-1);
}
```

(a) State **two** features of a recursive algorithm. [2 marks]

(b) State the value returned if the method is called by `recur(3)`. [2 marks]

12. Identify **two** reasons why the control system for a nuclear-power station would be required to use *real-time processing*. [2 marks]

13. (a) Express the decimal number **78** in *hexadecimal* format. [1 mark]

(b) Express the decimal number **–20** in 8-bit *two's complement* format. [2 marks]

14. Explain a possible benefit of the use of *defragmentation software*. [2 marks]

15. Define *polymorphism*, when used as a feature of object-oriented programming. [2 marks]

16. State the efficiency of a *quicksort* algorithm in BigO notation. [1 mark]

SECTION B

Answer **all** the questions.

- 17.** An application requires a list of names to be held in alphabetical order in the main memory. The program must allow for insertion or deletion of names from this list.
- (a) Explain why a *linked list* would be an appropriate data structure for holding these names. [3 marks]
 - (b) Describe the steps required to insert the name **Guillen** into the list. (It can be assumed that it will not be inserted either at the beginning or at the end.) [3 marks]
 - (c)
 - (i) Identify another data structure that would be suitable for maintaining this list of names. [1 mark]
 - (ii) For this new structure, discuss the advantages and disadvantages of maintaining a list of names over the original linked list. [3 marks]
- 18.** A company making car parts keeps a computerised file of records containing the name, code and price of each item it has for sale. *Direct access* will be used to access these records whenever a sale is made.
- (a) Suggest why direct access was chosen for this application. [2 marks]
 - (b) Explain how a *hashing algorithm* would be used in conjunction with the creation of this file. [2 marks]
 - (c) Explain why the hashing algorithm should be carefully chosen. [2 marks]

The hashing algorithm to be used is shown below.

$$\text{Address} = \text{key} \% 1000$$

- (d)
 - (i) Identify the record address generated for the key field 4231. [1 mark]
 - (ii) Suggest what would happen if, subsequently, a new record with the key field 6231 was added to the file. [3 marks]

19. An application is running on a computer. The main program calls up a subprogram. When the subprogram finishes, control is passed back to the main program.

- (a) Explain how a *stack* would be used to ensure that the correct sequence of instructions is followed. Reference should be made to the use of the *program counter*. [3 marks]
- (b) Explain why compilers convert mathematical expressions from *infix* into *postfix* notation. [2 marks]
- (c) Convert the infix expression $(A + B) / C$ into postfix. [2 marks]

Stacks can also be used when evaluating arithmetic expressions.

- (d) With the help of a diagram, explain how a stack would be used in the evaluation of the postfix expression $AB +$. [3 marks]

20. A small company has a LAN connecting its various desk-top computers and peripheral devices.

- (a) Explain, with an example, how *handshaking* might be used during data transmission over this LAN. [2 marks]

The company is going to provide Internet access to its LAN.

- (b) State the name of an additional hardware device that would be required to permit Internet access. [1 mark]
- (c) Explain how a firewall would help to provide security for the LAN. [3 marks]
- (d) Suggest, with reasons, **two** further measures that the company should take to safeguard its data from unlawful access via the Internet. [4 marks]

21. The method `logic1()`, shown below, returns the output from a particular *logic gate*. The logic gate has two inputs, *a* and *b*.

```
//parameters a and b can only have the values of 0 or 1
public boolean logic1(int a, int b)
{ if(!(a==b)) return true;
  else return false;
}
```

- (a) Identify the logic gate represented by the above method. [1 mark]
- (b) Construct the method `logic2()`, which would similarly return the output from a **NAND** gate. [2 marks]

Recall that two circuits are equivalent if their respective truth tables are the same.

The method `compareCircuits()` uses nested loops to generate and pass inputs to the methods `logic1()` and `logic2()`. It compares the outputs of the two logic circuits represented by these methods, and returns the value **true** if the two circuits are equivalent, and **false** if they are not.

- (c) Construct the method `compareCircuits()`. [5 marks]
- (d) Simplify the Boolean expression $\overline{A} \cdot \overline{B} + \overline{A} \cdot B + A \cdot \overline{B}$. [2 marks]

22. One of the functions of an *operating system* is file maintenance.

- (a) State two functions of *file maintenance*. [2 marks]
- (b) By using a diagram, identify the following parts of a multi-disk system: *track*, *sector*, *cylinder*. [3 marks]
- (c) Explain how the concept of *cylinders* can speed up the retrieval of data from a multi-disk magnetic disk system. [3 marks]
- (d) Explain why the technique of *blocking* is used when writing records onto a magnetic disk. [2 marks]