

MARKSCHEME

November 2007

COMPUTER SCIENCE

Higher Level

Paper 2

*This markscheme is **confidential** and for the exclusive use of examiners in this examination session.*

*It is the property of the International Baccalaureate and must **not** be reproduced or distributed to any other person without the authorization of IBCA.*

General Marking Instructions

*After marking a sufficient number of scripts to become familiar with the markscheme and candidates' responses to all or the majority of questions, Assistant Examiners (AEs) will be contacted by their Team Leader (TL). The purpose of this contact is to discuss the standard of marking, the interpretation of the markscheme and any difficulties with particular questions. It may be necessary to review your initial marking after contacting your TL. **DO NOT BEGIN THE FINAL MARKING OF YOUR SCRIPTS IN RED INK UNTIL YOU RECEIVE NOTIFICATION THAT THE MARKSCHEME IS FINALIZED.** You will be informed by e-mail, fax or post of modifications to the markscheme and should receive these about one week after the date of the examination. If you have not received them within 10 days you should contact your TL and IBCA. Make an allowance for any difference in time zone before calling. **AEs WHO DO NOT COMPLY WITH THESE INSTRUCTIONS MAY NOT BE INVITED TO MARK IN FUTURE SESSIONS.***

You should contact the TL whose name appears on your "Allocation of Schools listing" sheet.

Note:

Please use a personal courier service when sending sample materials to TLs unless postal services can be guaranteed. Record the costs on your examiner claim form.

General Marking Instructions

1. Once markscheme is received mark in pencil until final markscheme is received.
2. Follow the markscheme provided, do **not** use decimals or fractions and mark only in **RED**.
3. Where a mark is awarded, a tick (✓) should be placed in the text at the **precise point** where it becomes clear that the candidate deserves the mark.
4. Sometimes, careful consideration is required to decide whether or not to award a mark. Indeed, another examiner may have arrived at the opposite decision. In these cases write a brief annotation in the **left hand margin** to explain your decision. You are encouraged to write comments where it helps clarity, especially for moderation and re-marking.
5. Unexplained symbols or personal codes/notations on their own are unacceptable.
6. Record subtotals (where applicable) in the right-hand margin against the part of the answer to which they refer. Show a mark for each part question (a), (b), *etc.* Do **not** circle sub-totals. Circle the total mark for the question in the right-hand margin opposite the last line of the answer.
7. Where an answer to a part question is worth no marks, put a zero in the right-hand margin.
8. Record the mark awarded for each of the four questions answered in the Examiner Column on the cover Sheet.
Add up the marks awarded and enter this in the box marked TOTAL in the Examiner Column on the cover sheet.
9. After entering the marks on the cover sheet check your addition of all marks to ensure that you have not made an arithmetical error. Check also that you have transferred the marks correctly to the cover sheet. **We have script checking and a note of all clerical errors may be given in feedback to all examiners.**
10. Every page and every question must have an indication that you have marked it. Do this by **writing your initials** on each page where you have made no other mark.
11. A candidate can be penalized if he/she clearly contradicts him/herself within an answer. Once again make a comment to this effect in the left hand margin.

Subject Details: Computer Science HL Paper 2 Markscheme

Mark Allocation

Candidates are required to answer ALL questions (*[20 marks]* for question 1, *[20 marks]* for question 2, *[20 marks]* for question 3 and *[40 marks]* for question 4. Maximum total = *[100 marks]*).

General

A markscheme often has more specific points worthy of a mark than the total allows. This is intentional. Do not award more than the maximum marks allowed for part of a question.

When deciding upon alternative answers by candidates to those given in the markscheme, consider the following points:

- Each marking point has a separate line and the end is signified by means of a semi-colon (;).
- An alternative answer or wording is indicated in the markscheme by a “/”; either wording can be accepted.
- Words in (...) in the markscheme are not necessary to gain the mark.
- If the candidate’s answer has the same “meaning” or can be clearly interpreted as being the same as that in the mark scheme then award the mark.
- Mark positively. Give candidates credit for what they have achieved, and for what they have got correct, rather than penalising them for what they have not achieved or what they have got wrong.
- Remember that many candidates are writing in a second language; be forgiving of minor linguistic slips. Effective communication is more important than grammatical niceties.
- Occasionally, a part of a question may require a calculation whose answer is required for subsequent parts. If an error is made in the first part then it should be penalized. However, if the incorrect answer is used correctly in subsequent parts then **follow through** marks should be awarded. Indicate this with “**FT**”.

1. (a) 8; [1 mark]

(b) int/long; [1 mark]

(c) An example solution is:

```
public int countEmpty()
{
    // braces do not have to be used...
    int count = 0;
    for (int r = 0; r < 8; r++)
    {
        for (int c = 0; c < 8; c++)
        {
            if (board[r][c] == 0)
            {
                count = count + 1;
            }
        }
    }
    return count;
}
```

Award marks as follows up to [4 marks max].

a nested loop structure of any kind;
 loop with correct initial condition;
 loop with correct final condition;
 test of array element = 0;
 return of count or equivalent;

[4 marks max]

(d) An example solution is:

```
public void setBoard()
{
    // do even rows
    for (int r = 0; r < 7; r = r + 2)
        for (int c = 0; c < 7; c = c + 2)
            board[r][c] = -1;
    // do odd rows
    for (int r = 1; r < 8; r = r + 2)
        for (int c = 1; c < 8; c = c + 2)
            board[r][c] = -1;
}
```

Award marks as follows up to [4 marks max].

recognizing that two passes are necessary or an internal test of odd/even rows is needed (or some tricky mod/div combo is attempted) without being completely correct in implementation;
 correctly setting even rows, even columns;
 correctly setting odd rows, odd columns;
 setting board[r][c] to -1 for 16 distinct elements of the array (wherever!);

[4 marks max]

- (e) An example solution is:

```
public int testMove(int sr, int sc, int fr, int fc)
{
    // check it is not off the board
    if ( (fr < 0) || (fr > 7) || (fc < 0) || (fc > 7) )
        return 1;
    // check the move is one row, one column
    // (Math.abs could be used here)
    if (((fr - sr) != -1) && ( (fc - sc) != 1) || ((fc - sc) != -1))
        return 2;
    // check target square is empty
    if (board[fr][fc] > 0)
        return 3;
    // check it is not a -1 square
    if (board[fr][fc] < 0)
        return 4;
    // ok
    return 0;
}
```

Award marks as follows up to [8 marks max].

correct method signature;

each correct test **[4 max]**

each correct return code **[4 max]**

return of 0 if move succeeds;

[8 marks max]

Note that a (complete) solution like

```
if (board[fr][fc] == 0) return 0; (or return board[fr][fc];)
```

Does not follow the instructions given but may be awarded **[1 mark]** according to the above scheme.

- (f) *Award marks as follows up to [2 marks max].*

The –1 squares/half the board need not be stored;

Thus the (32 valid) positions could be stored in a linear structure (array/linked list);

Requiring some additional processing to convert to board positions;

Making it more complex to decide whether a move is valid;

[2 marks max]

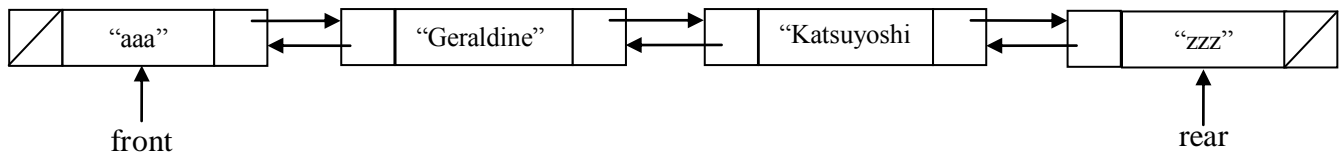
2. (a) Award marks for the following up to [2 marks max].

The Class contains both data members and methods;

The data members cannot be accessed directly/they are private;

The (external) behaviour of the class is defined by the (public) methods; [2 marks max]

- (b) An example diagram.



Award marks for the following up to [5 marks max].

both names in the nodes in the correct order;

the next pointer of “Geraldine” pointing to “Katsuyoshi” **and** the previous pointer pointing to “aaa”;

the previous pointer of “Katsuyoshi” pointing to “Geraldine” **and** the next pointer pointing to “zzz”;

front’s next pointer linked to “Geraldine”;

rear’s previous pointer pointing to “Katsuyoshi”;

[5 marks]

(c) An example solution is:

```
public void removeName()
{
    DoubleNode temp = front;
    boolean done = false;
    String name = input("Please input a name: ");
    // Move to first data node, check if list is empty
    temp = temp.getNext();
    if (temp.getName().equals("zzz"))
    {
        output("No names in this list");
    }
    else
    {
        // traverse to end
        do
        {
            if(temp.getName().equals(name))
            {
                // found it; remove is eased because there
                // will always be a previous and next node
                // link from previous to next then next to previous
                temp.getPrev().setNext(temp.getNext());
                temp.getNext().setPrev(temp.getPrev());
                // tidy up unlinked node references (not needed in answer)
                temp.setNext(null);
                temp.setPrev(null);
                // set exit loop flag
                done = true;
            }
            else
            {
                temp = temp.getNext();
            }
        } while ( !done && !(temp.getName().equals("zzz")) );
        // or !( done || (temp.getName().equals("zzz")) ) - de Morgan
        if (!done)
            output("Name does not exist in list");
    }
}
```

(Do not penalise if String comparisons use == rather than equals method)

Award marks as follows up to [10 marks max].

All local variables used are declared and initialized;

A name is input;

A good attempt is made to deal with the empty list case;

A correct attempt is made to deal with the empty list case **[2 marks]**.

An error message is output for an empty list;

The list is traversed in some way (move to next);

The traverse is stopped at the required node;

The previous node is linked to the next;

The next node is linked to the previous;

The traverse is terminated if the end of the list is found;

An error message is output when no name is found in the list;

[10 marks max]

(d) $O(1)$; **[1 mark]**

(e) *Award marks as follows up to [2 marks max].*

storing input characters;
for subsequent/sequential processing;
simulating a physical queue;
e.g. customers in a post office;
Accept any feasible application example.

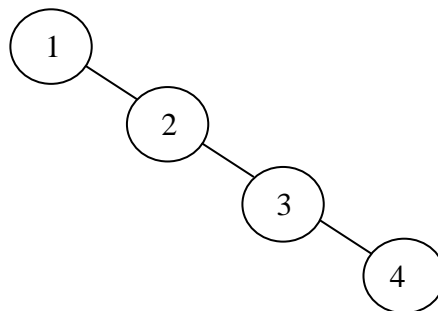
[2 marks max]

3. (a) *Award marks as follows up to [6 marks max].*

The root node of the tree is examined;
 the corresponding name is retrieved from the file;
 and compared to Ryan;
 if the name in the file is greater, traverse to the left;
 else traverse to the right;
 retrieve the (node) name;
 until a null pointer/node is located;
 (the record number for) Ryan should be inserted here;

[6 marks max]

- (b) Something like the following.



Accept if it contains names rather than numbers.

Award marks as follows up to [3 marks max].

4 nodes are linked in a line;
 1/Charles is the first node;
 4/Veronika is the last node;
 the drawing is completely correct;

[3 marks max]

- (c) *Award marks as follows up to [4 marks max].*

A/the well-balanced tree is more efficient;
 has an efficiency of $O(\log_2 n)$;
 the new tree is inefficient;
 has an efficiency of $O(n)$;
 the same as linear search;

[4 marks max]

- (d) *Award marks as follows up to [3 marks max].*

a full index could be used;
 this could be stored in a tree/other file/list;
 with record numbers in the order of ID;

[3 marks]

(e) *Award marks as follows up to [4 marks max].*

a partial index could be used;
this could be stored in a tree/other file/list;
not every name/ID need be entered (in the index);
the file needs to be changed;
so that records are in order;

[4 marks max]

4. (a) Award **[1 mark]** for a way and **[1 mark]** for an elaboration, for up to 3 ways, up to **[6 marks max]**.

Entering text into a computer/word processor;
by a blind person / person with limited dexterity;
using a microphone (as an input device);

Operating a wheelchair;
for a person who is unable to use other methods of control;

Operating a telephone dialler/application;
by a blind person / person with limited dexterity;
using a microphone (as an input device);
Any other reasonable outline.

[6 marks max]

- (b) Award **[1 mark]** for a way and **[1 mark]** for an elaboration up to **[4 marks max]**.

For example.

they could record their comments;
using a microphone/text input system;
which is converted to text output at the other computer;

they could type/scan parts of the book;
using OCR;
and add their own comments using a word processor;
then upload the file (to the other person's computer);

use sign language;
transmitted via the web cam;

they could hold up the book to the camera;
with comments typed into a chat session/shared word processor/etc;

[4 marks max]

- (c) *Award [1 mark] for a way and [1 mark] for an elaboration up to [4 marks max].*

For example:

using the web cam;
(with some way of explaining what each one means such as) text equivalents/letter/word cards;

recording the signs/image capture/video capture;
which are then put into a text doc/presentation;
together with explanations;

drawing the (sequence of movements associated with the) signs;
scanning them in;
and transmitting them to the other person;

[4 marks max]

- (d) (i) *Award [1 mark] for any of the following up to [2 marks max].*

Some (extreme/decorated) font faces may be hard to read;
for poorly sighted people;
small fonts should be avoided;
or the user should be allowed to change font / font face as required;
coloured fonts should be avoided;
as colour-blindness is relatively common among users;

[2 marks max]

- (ii) *Award [1 mark] for any of the following up to [2 marks max].*

Some people have limited dexterity;
and find scrolling difficult;
using a screen enlarger (for the poorly sighted);
is made more difficult (since even more scrolling is required);

[2 marks max]

- (iii) *Award [1 mark] for any of the following up to [2 marks max].*

People with hearing disabilities;
will not be able to (clearly) hear sounds;
they should not be used for important functions;
without some visual cue being added;

[2 marks max]

- (e) *Award marks for parts (i) and (ii) as follows.*
Identifying type of handicap [1 mark].
Stating the modification [1 mark].
Explaining its benefit [1 mark].

For example:

- (i) Partially sighted person would have troubling identifying/seeing the keys **[1 mark]**.
Could use a Braille keyboard **[1 mark]** that could be used by touch **[1 mark]**.
Person with dexterity motor control problems would have trouble operating the keyboard **[1 mark]**.
Could use keyboard with large keys **[1 mark]** that would allow greater control **[1 mark]**.
[3 marks max]
- (ii) Person without use/control of hands **[1 mark]**, could use a pointer that could be attached to the head / head mouse **[1 mark]**, head movement will move the pointer **[1 mark]**, or could use touch pads **[1 mark]** that are easy/easier to manipulate **[1 mark]**.
[3 marks max]

- (f) *Award marks as follows up to [4 marks max].*

They can read across columns/frames **[1 mark]**, giving nonsense/garbled information **[1 mark]**.
They can read all unnecessary information **[1 mark]**, slowing down the process **[1 mark]**.
[4 marks max]

- (g) *Award [1 mark] for a suitable way and [1 mark] for a justification/explanation.*

For example:

- (i) any sounds would need to be replaced/augmented;
with visual cues or text; **[2 marks]**
- (ii) where colour is being used for emphasis;
it should be checked, (eg using an equivalent greyscale image);
and altered (to greyscale) as appropriate; **[2 marks max]**
- (iii) parts requiring interaction (eg menus, buttons);
should be made large/easy to access; **[2 marks]**

- (h) *Award marks as follows up to [4 marks].*

sound is analogue data;
it needs to be converted;
to digital form;
for (computer) processing; **[4 marks]**
