IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

N07/5/COMSC/HP2/ENG/TZ0/XX

88077012

COMPUTER SCIENCE
HIGHER LEVEL
PAPER 2

Tuesday 13 November 2007 (morning)

2 hours 15 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Answer all the questions.

© IBO 2007

*Answer **all** the questions.*

1. An *array* is used to represent the status of a board game. In the representation of the board, below left:

- The board is 8 squares by 8 squares
- The game is played only on the white squares, the dark squares cannot be moved to
- The 'x' symbol represents player 1's pieces
- The 'o' symbol represents player 2's pieces

The Board

	x		x		x		x
x		x		x		x	
	x		x		x		x
o		o		o		o	
	o		o		o		o
o		o		o		o	

The Array

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
[0]	-1	1	-1	1	-1	1	-1	1
[1]	1	-1	1	-1	1	-1	1	-1
[2]	-1	1	-1	1	-1	1	-1	1
[3]	0	-1	0	-1	0	-1	0	-1
[4]	-1	0	-1	0	-1	0	-1	0
[5]	2	-1	2	-1	2	-1	2	-1
[6]	-1	2	-1	2	-1	2	-1	2
[7]	2	-1	2	-1	2	-1	2	-1

In the array representation of the game:

- –1 represents a square that cannot be moved to by either player
- 0 represents a white square that is currently empty
- 1 represents a square occupied by player 1
- 2 represents a square occupied by player 2

(a) State the number of empty white squares shown in the above diagram. *[1 mark]*

(b) State a suitable data type for an element of the array. *[1 mark]*

(c) Construct the *algorithm* that calculates and returns the number of empty squares on the board. *[4 marks]*

(This question continues on the following page)

(Question 1 continued)

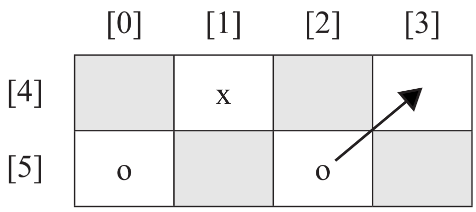
When the program begins the board array needs to be initialized with the values shown in the array diagram on the previous page. For example, all of the elements corresponding to the squares that cannot be moved to should be set to the value –1.

(d) Construct a method that sets to –1 all of the array elements that cannot be moved to. *[4 marks]*

Consider the player playing the o pieces. He may only move a piece following these rules:

1. The piece cannot move to a row or column that is off the edge of the board.
2. The piece must move forward 1 row and to the side 1 column only (i.e. diagonally, 1 square).
3. The piece cannot move to a position occupied by another piece (of either player).
4. The piece cannot move to a square represented by a –1.

In the diagram given below (showing a small part of the board), the arrow represents the only available move that follows the 4 rules given above.



(e) Construct the method `testMove` that accepts 4 parameters as follows:

- the row and column subscripts/indexes of the piece to be moved
- the row and column subscripts/indexes that the player wishes to move to

and returns a numeric code from 1 to 4 if one of the 4 rules, stated above, is broken.

For example:

```
int theMove = testMove(5, 2, 4, 3)
```

and a 0 would be returned to indicate that the move should succeed. *[8 marks]*

(f) Outline a method of storing the status of the board, which uses less memory. *[2 marks]*

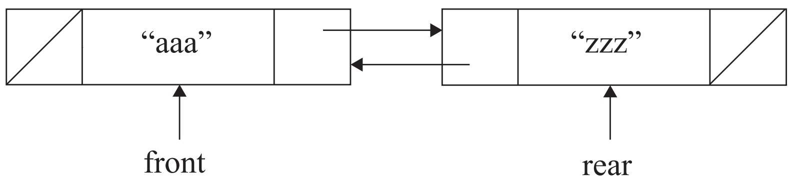
2. The following code represents a *class* that can be used to create a *doubly-linked list*:

```
public class DoubleNode
{
  private String name;
  private DoubleNode next;
  private DoubleNode prev;

  public DoubleNode()
  {
    name = null;
    next = null;
    prev = null;
  }
  public DoubleNode(String nm, DoubleNode nx, DoubleNode pv)
  {
    setName(nm);
    setNext(nx);
    setPrev(pv);
  }
  public void setName(String n){ name = n; }
  public void setNext(DoubleNode n){ next = n; }
  public void setPrev(DoubleNode p){ prev = p; }
  public String getName(){ return name; }
  public DoubleNode getNext(){ return next; }
  public DoubleNode getPrev(){ return prev; }
}
```

(a) Explain the concept of *encapsulation* using the above code as an example. *[2 marks]*

The DoubleNode Class is used to create a list of names with front and rear pointers to special nodes:



(This question continues on the following page)

(Question 2 continued)

The following statements are then executed:

```
DoubleNode d = new DoubleNode();
d.setName("Katsuyoshi");
DoubleNode x = new DoubleNode("Geraldine", d, front);
d.setPrev(x);
front.setNext(x);
rear.setPrev(d);
d.setNext(rear);
x.setPrev(front);
```

(b) Draw and label a diagram of the resulting doubly-linked list. *[5 marks]*

(c) Construct the code that does the following:

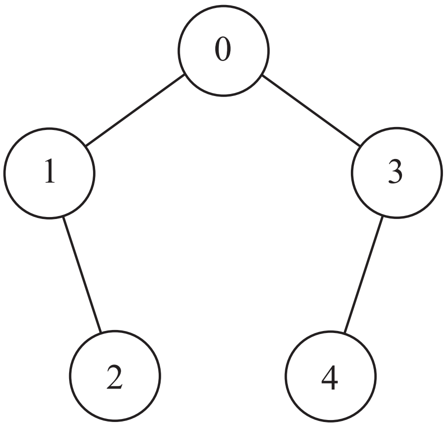
- inputs a name
- traverses the list
- if the list has no data nodes outputs an error message
- if the name is not in the list outputs an error message
- otherwise deletes the node containing the name from the list

There are no duplicate names in the list, the "aaa" and "zzz" nodes are always present. *[10 marks]*

(d) State the *BigO* efficiency of the algorithm outlined in part (c). *[1 mark]*

(e) Outline an application that could use a *queue*. *[2 marks]*

3. Consider the following diagrams of a *binary tree* and a *file*:



	ID	Name	Address
[0]	34	Hamza	11 Arctic Circle
[1]	21	Anton	25 Major Road
[2]	67	Binh	53 Triumph Avenue
[3]	22	Nikki	84 Cloud Square
[4]	19	Maggie	88 Lucky Street

The rows in the table represent records in the file and each node in the binary tree contains the position of a given record in the file.

When a new record is added to the file, it is added at the end of the file and the record number is inserted in the correct position in the tree, such that an in order traversal of the tree visits the nodes in alphabetical order of the name.

A record for Ryan is added to the file.

(a) Explain the steps required to locate the correct position in the binary tree for this new record. *[6 marks]*

A new tree and file are started and the following sequence of names is added:

Charles
Chervonne
Naomi
Veronika

(b) Draw a sketch of the resulting binary tree. *[3 marks]*

(c) Discuss the efficiency of access of the new tree, created in part (b), compared to the original tree. *[4 marks]*

The file contains other data such as a unique ID and address information.

(d) Explain how the names could also be output in order of ID number (as well as the existing name order). *[3 marks]*

The file is expanded to contain over 1000 records.

(e) Explain how the file could be re-organized so that the tree need not contain a node for every record in the file. *[4 marks]*

This question requires the use of the Case Study.

4. (a) Outline **three** ways in which voice recognition systems can assist disabled persons. *[6 marks]*

Two people are hearing impaired and cannot use the telephone. However they both have computers equipped with standard office software and web cameras.

(b) Outline **two** ways by which they could discuss ideas (using the computer system) related to a book they are both reading. *[4 marks]*

One of them knows sign language and is teaching the other basic signs.

(c) Explain **two** ways by which instruction could be given, using the computer system. *[4 marks]*

(d) Discuss possible problems for people with disabilities of the use of the following in web site design:

(i) Unusual font faces and sizes *[2 marks]*

(ii) Scrolling (horizontal or vertical movement required to view a page) *[2 marks]*

(iii) Sounds *[2 marks]*

(e) For the following two hardware devices, identify a problem that a disabled person might have in using it, and suggest a possible modification that might help that person.

(i) keyboard to enter data *[3 marks]*

(ii) mouse to select a menu option *[3 marks]*

(f) Explain **two** problems that might arise when a screen reader reads text from a normal web page. *[4 marks]*

(This question continues on the following page)

(Question 4 continued)

A software engineer is producing a new graphical package.

(g) Describe ways in which an interactive multimedia tutorial might need to be modified so that it would be suitable for:

(i) A person with hearing impairment *[2 marks]*

(ii) A person who is colour blind *[2 marks]*

(iii) A person who lacks fine motor control. *[2 marks]*

(h) Explain the role of *Analog-digital conversion* in the operation of a voice-activated wheelchair. *[4 marks]*