N11/5/COMSC/HP2/ENG/TZ0/XX

International Baccalaureate®
Baccalauréat International
Bachillerato Internacional

COMPUTER SCIENCE
HIGHER LEVEL
PAPER 2

Friday 18 November 2011 (morning)

2 hours 15 minutes

INSTRUCTIONS TO CANDIDATES

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

*Answer **all** the questions.*

1. Consider the following class.

```
class Animal
{
  public int animalID;
  public String animalSpecies;
  ...
}
```

Using this class, an object can be created for each individual animal. The object contains an identifier for the animal and identifies what species the animal is.

Linked lists are to be used to hold groups of `Animal` objects.

(a) Construct a code modification to the `Animal` class that allows `Animal` objects to be linked together in a list. *[1 mark]*

(b) Describe, with the aid of a diagram, how objects of the `Animal` class can be used to construct a *linked list* of `Animal` objects. The first object in the list is pointed to by `animalList`. *[3 marks]*

The class `Zoo`, shown below, maintains an inventory of the animals in a zoo. The linked list `animalList` contains one `Animal` object for each animal in the zoo. The zoo has six species of animals and the species' names are contained in the array `allSpecies`.

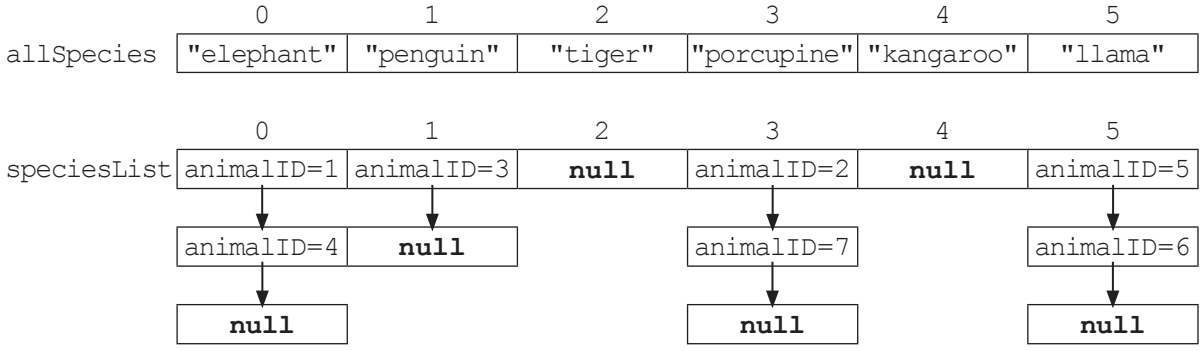
```
public class Zoo
{
  // instance variables
  private Animal animalList;  // points to first element of linked list
  private String[] allSpecies = {"elephant", "penguin", "tiger",
                                 "porcupine", "kangaroo", "llama"};
  private Animal[] speciesList = new Animal[6];

  // constructor for objects of class Zoo
  public Zoo()
  {
    // animalList is initialized with Animal objects
    // one object for each animal in the zoo
    // speciesList is initialized with each element set to null
  }

  public int getSpeciesIndex()
  {
    // lines of code missing
  }

  public void divideAnimals()
  {
    // lines of code missing
  }

  int speciesCount(String species)
  {
    // lines of code missing
  }
}
```

(This question continues on the following page)

(Question 1 continued)

(c) Construct the method `getSpeciesIndex()` that, when passed the name of a species, will return the index of the element in `allSpecies` which contains that name. *[4 marks]*

The `Animal` objects in `animalList` are to be divided amongst separate linked lists, one list for each species. The first object in each of these separate lists (or **`null`** if the list is empty) is held in an array that corresponds to the `allSpecies` array, as shown below.



(d) Assuming that the `Animal` objects have already been divided into the linked lists in the array `speciesList`, construct the `speciesCount()` method that finds the linked list corresponding to the species named in the argument and returns the total number of animals found in that list. You may use the `getSpeciesIndex()` method. *[4 marks]*

(e) Construct the method `divideAnimals()` that divides the objects in the list `animalList` into the separate linked lists in `speciesLists` corresponding to the different species in `allSpecies`. You may use the `getSpeciesIndex()` method. *[8 marks]*

2. Scientists are monitoring the fish population in a river. They have installed an automated system that counts how many fish swim past a sensor every hour. The system sends an automatic text message to the lab containing the hour and the number of fish counted during the previous hour. The received text messages are processed by a computer program running in the lab.

Example of message:
06, 523

The above message was sent at 06:00. In the previous hour 523 fish swam past the sensor.

The received data is processed by the class `FishCounts`, partially shown below. The counts are stored in an array named `counts`. The data received for the hour starting at 00:00 is stored in `counts[0]`, the data for the hour starting at 01:00 is stored in `counts[1]` *etc*.

```
class FishCounts
{
  public int[] counts = new int[24];

  public FishCounts()
  {
    for (int i = 0; i < 24; i = i + 1)
    {
      this.counts[i] = -1;
    }
  }

  public void fillMissingCounts()
  {
    // lines of code missing
  }

  public FishStats findTopTwo()
  {
    // lines of code missing
  }
}
```

However, not all text messages are received. In these cases the data is represented by setting the corresponding element of `counts[]` to a negative value.

(a) Explain why the class `FishCounts` contains a method called `FishCounts()` and what the method accomplishes. *[3 marks]*

(This question continues on the following page)

(Question 2 continued)

The data is processed at the end of each day. When the count for a given hour is missing, it can be estimated as the average of the count from the previous hour and the following hour.

(b) Construct the method `fillMissingCounts()` that is run at the end of the day and replaces any negative values in `counts[]` with the average of the previous value and the following value. You may assume that the first and last values of `counts[]` are not negative and that there are no two consecutive hours where data is missing. *[4 marks]*

The data in `counts[]` is to be analysed and the results put into an object of the `FishStats` class.

```
class FishStats
{
  public int max;        // the maximum count
  public int maxIndex;   // the index at which the maximum count occurs
  public int min;        // the minimum count
  public int minIndex;   // the index at which the minimum count occurs
  public int max2;       // the second largest count
  public int mean;       // the average count
  ...
}
```

(c) Using an example, explain why representing all of the variables as **`int`** could lead to inaccuracies in the mean. *[3 marks]*

(d) In the `FishCounts` class, construct a method `findTopTwo()` which finds the largest two values in `counts[]`, creates a new `FishStats` object and updates `max`, `maxIndex`, `max2` and `mean` before returning it. Note that `max2` may be the same as `max`, if the maximum count occurred in two or more hours. *[10 marks]*

3. A refugee camp was set up by a relief organization after a large-scale disaster. The camp maintains a file called `refugees` containing a list of the names, ages, gender and former addresses of the refugees who are living there. When relatives contact the camp seeking a particular person, the `refugees` file is searched for records matching the person's name.

(a) Define the term *record*. *[2 marks]*

(b) One method of organizing the `refugees` file could be with the use of a *partial-index*.

(i) Explain the steps needed to construct a partial-index to make searching the file more efficient. *[3 marks]*

(ii) Describe the process of searching the file using the partial-index. *[2 marks]*

The number of refugees in the camp is continually changing as new refugees arrive and others are reunited with their families.

(c) Explain why a partial-index is not the best choice of file organization for this application. *[3 marks]*

The refugee camp is one of a number of refugee camps run by a relief organization in the disaster area. At its headquarters, the relief organization has decided to use a master file which maintains an updated list of the refugees in all of the camps. Each camp sends a transaction file to the headquarters on a regular basis.

(d) Suggest how the *key field* for the transaction files would be created. *[3 marks]*

(e) Describe the content and organization of the files sent by the camps. *[2 marks]*

(f) Describe the process by which the master file of refugees is updated. *[5 marks]*

4. *This question requires the use of the case study.*

(a) Each component system in Heathrow Terminal 5 (T5) was tested before the terminal was opened to the public. A variety of testing methods were used.

(i) Define the term *parallel running*. *[2 marks]*

(ii) Explain why simulations were used in testing the new baggage handling system. *[4 marks]*

(iii) Identify **two** factors that should be considered when selecting a testing method. *[2 marks]*

(b) Ground services in T5 include a fleet of electric carts that help disabled passengers to move between different parts of the airport. Each cart has an onboard computer that connects with the airport computer system to indicate its location and to receive instructions.

(i) Outline a suitable method of data communication between carts and the airport system. *[2 marks]*

(ii) Describe **two** potential problems that could arise from this method of implementation. *[4 marks]*

(iii) Describe **two** ways in which the data communication between the carts and the airport computer system could be made secure. *[4 marks]*

(c) Air traffic controllers continue to resist efforts to replace the current method of paper strips for each flight.

(i) With specific reference to the case study, identify **two** features of the paper strips method that air traffic controllers like. *[2 marks]*

(ii) Suggest a way in which a computer-based system could provide the two features you identified in part (i). *[4 marks]*

(iii) Discuss whether the safety concerns associated with replacing the paper strips method with a computer-based system can be overcome. *[6 marks]*

(This question continues on the following page)

(Question 4 continued)

(d) A major problem in airports is the length of queues at check-in and at passport control. The airport management is investigating possible solutions to resolve the problem.

(i) State **two** ways to uniquely identify a person in the airport. *[2 marks]*

(ii) Discuss how technology can be used to manage the queues at check-in more effectively. *[4 marks]*

(iii) Describe **two** privacy or ethical concerns that might be raised by the method(s) you discussed in part (ii), above. *[4 marks]*