Sample test

Algorithm Design and Data Structures
COMPSCI 1103, 2103

Official Reading Time:	10 mins
Writing Time:	120 mins
Total Duration:	130 mins

Questions	Time	Marks
Answer all 7 questions	120 mins	120 marks
		120 Total

Instructions

- Begin each answer on a new page in the answer book.

- Examination material must not be removed from the examination room.

Materials

- Foreign language paper dictionaries permitted.

DO NOT COMMENCE WRITING UNTIL INSTRUCTED TO DO SO

Programming Fundamentals

Question 1

(a) You use the `new` keyword to allocate a dynamic array.

    i. What's the difference between a dynamic array and an ordinary array?

[2 marks]

    ii. Where does the memory for the dynamic array come from?

[1 mark]

    iii. With a dynamic array, you need to call `delete[] array` to deallocate the memory. What will happen if you omit the square brackets?

[2 marks]

(b) Please consider each of the statement carefully and give the answer **true** or **false** and justify your answer.

    i. The elements in a linked list are contiguous in memory

[2 marks]

    ii. The pointer pointing to a dynamic variable is always stored on the stack.

[2 marks]

(c) What is the output of the following code fragment?

```
int* x, y;
int arr[5] = {1, 2, 3, 4, 5};
x = &arr[0];
y = arr[0];
cout << *x << " " << y << endl;
*x = 10;
cout << *x << " " << y << endl;
x = x+2;
y = arr[0];
cout << *x << " " << y << endl;
```

[3 marks]

(d) Write a loop that will search an array of `int` values, `myArray`, to determine if it contains the integer value stored in a variable `myInt` and display the message "Found" if it finds it.

Clearly comment your code. You may assume the `myArray` and `myInt` are both declared and initialised.

[6 marks]

[Total for Question 1: 18 marks]

Please go on to the next page. . .

Inheritance and Object Oriented Programming

Question 2

(a)   i. What is inheritance?

[2 marks]

ii. Using an example, explain how inheritance facilitates code reuse.

[4 marks]

(b) Define polymorphism, and explain what the `virtual` keyword does when added to a function.

[3 marks]

(c) Please clearly describe, in the context of C++ inheritance, the difference between redefining, overloading and overriding, using diagrams where necessary.

[6 marks]

(d) A rock paper scissors (RPS) game consists of two Players of type `Player`. A separate class called `BestOfThreeReferee` is used to compare `Player` moves. `Player` is defined with the following interface:

```
class Player{
private:
  string name;
public:
  Player(string name);
  string getName();
  string performMove();
};
```

`BestOfThreeReferee` has three separate pieces of functionality it must perform. It must be able to determine if a move is valid. It must be able to take in two moves and return the winning move. It must be able to have two Players fight, taking in two Player pointers and returning the name of the winner.

i. Provide the interface for `BestOfThreeReferee`.

[3 marks]

ii. The fight function takes in two `Player` pointers, makes them play three rounds, and returns the name of the `Player` with the most wins. If the `Players` have the same number of wins, `fight` returns an empty string. Implement the fight function of BestOfThreeReferee.

[6 marks]

[Total for Question 2: 24 marks]

Please go on to the next page. . .

Recursion

Question 3

(a) What are the three requirements for successful recursion in C++?

[3 marks]

(b) Please explain the advantages and disadvantages of using recursion instead of an iterative approach.

[2 marks]

(c) Why does a recursive function use the stack? Please use an example to show how the stack is used in recursion.

[4 marks]

(d) The *digital root* of a number is defined as the single-digit number obtained by recursively adding all the digits in the number.
For example,

- The digital root of 12345 is 6, since $1 + 2 + 3 + 4 + 5 = 15$, and $1 + 5 = 6$.
- The digital root of 43265132 is 8, since $4 + 3 + 2 + 6 + 5 + 1 + 3 + 2 = 26$, and $2 + 6 = 8$.

Please write a recursive function `digitalRoot` that takes an `int` n and returns the digital root of n.

[10 marks]

[Total for Question 3: 19 marks]

Algorithmic Strategies

Question 4

    (a) Give an example of a brute force strategy and where you might use it.

[4 marks]

    (b) What is a Divide-and-Conquer strategy? Give an example.

[4 marks]

    (c) What is a Greedy strategy? Give an example.

[4 marks]

[Total for Question 4: 12 marks]

Complexity Notation

Question 5

    (a) What is the definition of $f(n)$ being in $O(g(n))$?

                    [1 mark]

    (b) What is the definition of $f(n)$ being in $\Omega(g(n))$?

                    [1 mark]

    (c) What is the definition of $f(n)$ being in $\Theta(g(n))$?

                    [1 mark]

    (d) Please prove that $n^4 + 3n^2 + 9$ is in $\Theta(n^4)$.

                    [4 marks]

    (e) Please prove that $n^4 + 3n^2 + 9$ is not in $O(n^3)$.

                    [1 mark]

    (f) Given that $f(n) \in O(n)$ and $g(n) \in O(\log n)$, please prove that $f(n)+g(n) \in O(n)$.

                    [4 marks]

[Total for Question 5: 12 marks]

Sorting and Searching

Question 6

   (a) Please illustrate the process of merging the two sorted list $\{3, 5, 9, 10\}$ and $\{9, 12, 14, 15\}$ in mergesort.

[2 marks]

   (b)   i. Given a list of n integers, you are asked to sort them in **descending** order using *quicksort*. Please write down the pseudo code of quicksort.

[5 marks]

        ii. What is the worst-case performance of quicksort?

[1 mark]

       iii. The performance of quicksort depends on the selection of the pivot value. What kind of pivot value will result in the worst-case performance?

[2 marks]

       iv. What is the best-case performance of quicksort?

[1 mark]

       v. What kind of pivot value will result in the best-case performance?

[2 marks]

   (c) Given a list of n `int` (sorted, in **ascending** order), please provide the pseudo code of binary search to find out whether the `obj` value is in the list.

[4 marks]

[Total for Question 6: 17 marks]

Linked Lists

Question 7

    (a) Define a linked list containing n nodes as follows:

```
struct Node {
  int data;
  Node *link;
}
```

       What is the complexity for deleting a node at the end of the linked list? Please also provide the pseudo-code.

                                                                [5 marks]

    (b) Stacks and Queues are often implemented based on linked lists.

         i. What is a stack?

                                                              [1 mark]

       ii. What is a queue?

                                                              [1 mark]

      iii. What are the common operations of the stack?

                                                           [2 marks]

    (c) What is the complexity for adding a node in the middle of the linked list?

                                                               [2 marks]

    (d) A binary tree is a tree with at most 2 child nodes.

        i. Please provide the C++ code for a node in a binary tree.

                                                            [2 marks]

       ii. An important application of binary trees is the use in searching. Please provide pseudo code for the search function in a binary search tree.

                                                         [5 marks]

                                             [Total for Question 7: 18 marks]

End of exam