PAPER 4: PROGRAMMING

COMPUTER SCIENCE

Topic: Paper 4: Programming

INSTRUCTIONS

- Carry out every instruction in each task.
- Answer **all** questions.
- Use a black or dark blue pen.
- You may use an HB pencil for any diagram, graphs or rough working.
- **Calculator Not Allowed**.
- Show your workings if relevant.

INFORMATION

- The number of marks for each question or part question is shown in brackets [ ].

A class declaration can be used to declare a record.
If the programming language used does not support arrays, a list can be used instead.

---

1. A programmer is trying to create a program which will use n-gram model to fill in the blanks. For this, he is planning to start by calculating frequency of words in a paragraph.

(a) The generated words and their frequency will be stored in two arrays: 'Words' and 'Count'.
Write program code to declare 'Words' and 'Counts'. [1]

File 'Text.txt' contains text about Sir Issac Newton.

(b) The procedure 'Readfile()':

- Opens the file 'Text.txt'
- Reads content of the file
- Raises an exception if the file is not found
- Adds words in Words and their count in Counts
- Returns the total number of words read from file

Write program code for Readfile(). [10]

The programmer later discovered that a sorted list of words(alphabetically) would be better.

(c) The procedure 'Sort()' uses Insertion sort to sort the 'Words' array and the corresponding counts.
Write program code for 'Sort()'. [7]

(d) The main program does the following:

- Calls Readfile()
- Displays first 10 items of Words along with their count.
- Calls Sort()
- Displays first 10 items of Words along with their count.

Ammend the main program to do that. [5]

He decided that sorting based on count would be better.

(e) Ammend 'Sort()' to sort based on count instead [2]

(f) Test 'Sort()' and save screenshot in evidence doc. [2]

2. A computer game is being developed.
The game is written using object oriented programming.
The class GameObject stores data about entities in the game.

GameObject	
x : REAL	stores the x-coordinate
y : REAL	stores the y-coordinate
SpeedX : REAL	stores the speed along x-axis
SpeedY : REAL	stores the speed along y-axis
Constructor()	initialises SpeedX,SpeedY to its parameter values and x and y coordinate to 0
GetX()	returns x
GetY()	returns y
GetSpeedX()	returns SpeedX
GetSpeedY()	returns SpeedY
SetX()	sets x to parameter value
SetY()	sets y to parameter value
SetSpeedX()	sets SpeedX to parameter value
SetSpeedY()	sets SpeedY to parameter value
DisplayInfo()	displays Information about the object (x,y,SpeedX and SpeedY) in organized way.
Update()	updates x and y based on SpeedX and SpeedY based on time taken as parameter

(a) Write program code to declare the class GameObject and its constructor. [4]

(b) The get methods GetX(), GetY(), GetSpeedX() and GetSpeedY() each return relevant attribute.
Write program code for these methods. [4]

(c) The set methods SetX(), SetY(), SetSpeedX() and SetSpeedY() each set relevant attribute to parameter value.
Write program code for set methods. [4]

(d) The method Update() updates x and y based on SpeedX and SpeedY.
It takes dt as parameter which is the time taken.
Write program code for Update(). [4]

(e) Write program code for function DisplayInfo [2]

(f) Write main program that:
Creates a new object of class GameObject.
Sets x to 10, y to 20, SpeedX to 5 and SpeedY to 10.
Updates the object with time taken as 2.
Call DisplayInfo. [4]

(g) A subclass named Enemy that inherits from the GameObject class.
The Enemy class should have an additional attribute health.
Override the DisplayInfo method from the parent class to include health information.

Enemy	
health	An additional attribute to store the health of the enemy.
DisplayInfo()	Overrides the parent class method to include health information.
TakeDamage(damage)	Method to subtract damage from health. If health falls below 0, set it to 0.

Write program code for declaring class Enemy along with its constructor. [4]

(h) Write program code for DisplayInfo() method. [2]

(i) Write program code for method TakeDamage [2]

(j) Create a method TakeDamage in the Enemy class that takes damage as a parameter and subtracts it from the health.
If the health falls below 0, set it to 0. [4]

(k) Create an enemy object with value of your choice. Call the TakeDamage method on the enemy object with damage=25
Display the updated information of the enemy object using the DisplayInfo method. [2]

3. A programmer needs to manipulate strings and numbers for a text processing application.

(a) Write a function **reverse_string(s)** that takes a string **s** as input and returns the string reversed. [3]

(b) Write a function **is_palindrome(s)** that takes a string **s** as input and returns **True** if the string is a palindrome, and **False** otherwise. [3]

(c) Write a function **sum_of_digits(n)** that takes an integer **n** as input and returns the sum of its digits. [3]

(d) Write a function **count_vowels(s)** that takes a string **s** as input and returns the number of vowels in the string. [1]

(e) Test the above function with appropriate arguments. [2]