Cambridge Assessment International Education

Cambridge International AS & A Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

1596084089

COMPUTER SCIENCE **9608/42**

Paper 4 Further Problem-solving and Programming Skills **May/June 2020**

2 hours

You must answer on the question paper.

No additional materials are needed.

INSTRUCTIONS

- Answer **all** questions.
- Use a black or dark blue pen.
- Write your name, centre number and candidate number in the boxes at the top of the page.
- Write your answer to each question in the space provided.
- Do **not** use an erasable pen or correction fluid.
- Do **not** write on any bar codes.
- You may use an HB pencil for any diagrams, graphs or rough working.
- Calculators must **not** be used in this paper.

INFORMATION

- The total mark for this paper is 75.
- The number of marks for each question or part question is shown in brackets [ ].
- No marks will be awarded for using brand names of software packages or hardware.

This document has **16** pages. Blank pages are indicated.

© UCLES 2020

1 Amar is alerted to a run-time error when he runs a program.

(a) A run-time error occurs when Amar attempts to open a file that does not exist.

State **three other** reasons why a run-time error may occur.

1 .....

.....

2 .....

.....

3 .....

.....

[3]

(b) A program should be written with exception handling routines to manage run-time errors.

A program reads data from the text file `MyData.txt`. The program needs to report an exception if it attempts to open the file and the file does not exist.

Write **program code** to handle and report this exception.

Programming language .....

Program code

.....

.....

.....

.....

.....

.....

.....

.....

..... [2]

2 A business is developing a program that stores each customer's username and password in a hash table.

The hash table will be implemented as a 1D array, `CustomerLogIn`, of the custom data type `CustomerRecord`.

The declaration for `CustomerRecord` is:

```
TYPE CustomerRecord

      DECLARE Username : STRING

      DECLARE Password : STRING

ENDTYPE
```

The hash table will store a maximum of 3000 records. The key field will be `Username`.

(a) The program declares the hash table and initialises the username and password of all the records to an empty string.

Write **pseudocode** to declare **and** initialise the hash table.

.....

.....

.....

.....

.....

.....

.....

..... [4]

(b) (i) A function, `SearchHashTable()` will search for a customer's record in the hash table.

The function `Hash()`:

- takes a `Username` as a parameter
- performs the hashing algorithm
- returns the calculated index of the username within the hash table.

The function `SearchHashTable()`:

- takes the username to search for as a parameter
- uses `Hash()` to calculate the first index of this username within the hash table
- returns either the index of the username if found, or –1 if not found.

Complete the **pseudocode** for the function `SearchHashTable()`.

```
FUNCTION SearchHashTable(BYVALUE SearchUser : STRING) RETURNS .....
    DECLARE Index : INTEGER
    DECLARE Count : INTEGER
    Index ← ..... (SearchUser)
    Count ← 0
    WHILE (CustomerLogIn[Index] ..... <> .....)
           AND(CustomerLogIn[Index].Username <> "")
           AND(Count < 2999)
        Index ← Index + 1
        Count ← Count + 1
        IF Index > .....
           THEN
              Index ← 0
        ENDIF
    ENDWHILE
        IF CustomerLogIn[Index].Username = .....
           THEN
              RETURN Index
           ELSE
              RETURN .....
        ENDIF
    ENDFUNCTION
```

[7]

(ii) Explain the purpose of the variable `Count` in the function `SearchHashTable()`.

.....

.....

.....

..... [2]

3 Recursive algorithms can be used when creating programs.

(a) Describe what is meant by a **recursive algorithm**.

.....

.....

.....

.....

.....

..... [3]

(b) A string that is a palindrome, reads the same forwards as it does backwards. For example, the name Anna is a palindrome.

The function `Substring(Variable, StartingCharacter, NumberOfCharacters)` returns one or more characters from a string. The first character is at position 0.

For example, the string `"Happy"` is stored in the variable `Word`.

- `Substring(Word, 1, 1)` would return the character `"a"`.
- `Substring(Word, 2, 3)` would return the characters `"ppy"`.

The function `Length()` returns the length of the string as an integer. For example, `Length(Word)` returns `5`.

The following is a recursive function to find out whether a string is a palindrome. The function returns `True` if the parameter is a palindrome, and returns `False` if it is not a palindrome.

Complete the **pseudocode** for the recursive algorithm to indicate whether a string is a palindrome.

```
FUNCTION IsPalindrome(CheckWord : STRING) RETURNS BOOLEAN
    IF ..... <= 1
        THEN
            RETURN .....
    ENDIF
    IF Substring(CheckWord, 0, 1) <>
             Substring(CheckWord, ..... (CheckWord)-1, 1)
        THEN
            RETURN .....
        ELSE
            RETURN .....(Substring(CheckWord, 1,
                                   Length(CheckWord)-2))
    ENDIF
ENDFUNCTION
```

[5]

(c) The function `FindPower()` is a recursive function that calculates the result of a base number to the power of the exponent. For example, the result of $2^4 = 16$, as 2*2*2*2 = 16. In this example, 2 is the base number and 4 is the exponent.

The base number and the exponent are passed as parameters.

Write **pseudocode** for the recursive function `FindPower()`. Assume both the base and the exponent are positive integers.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

4 **(a)** A tennis club has a booking form to book lessons with an instructor.

Club members can book up to five lessons using the booking form.

The customer details section has the data:

- name
- address
- telephone number.

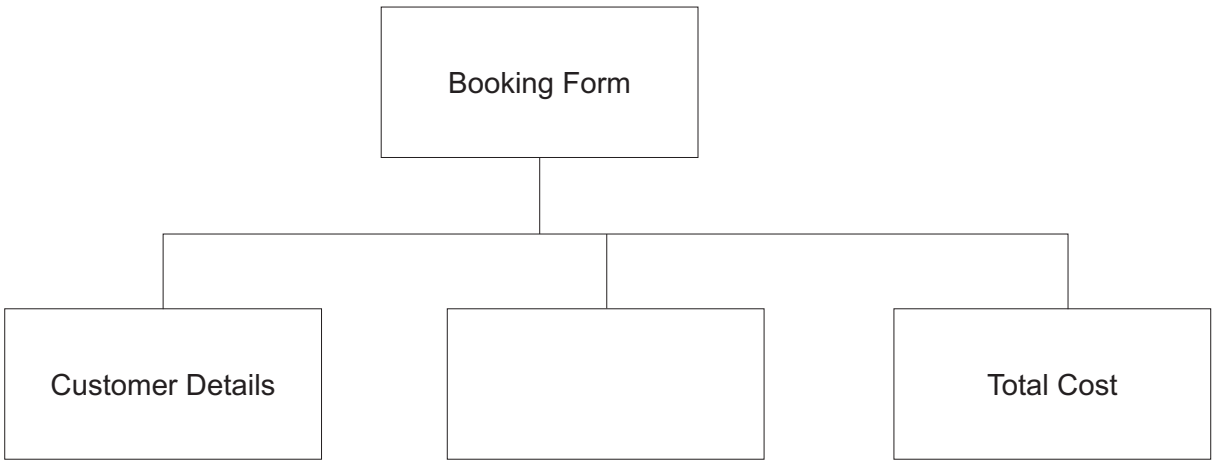
The lesson details section has the data:

- lesson type
- date and time
- lesson cost.

The cost of each lesson is dependent on the customer's type of membership. The membership can be bronze, silver or gold.

The total cost is also calculated.

Complete the following JSP data structure diagram for the booking form.



[7]

(b) State **two** programming constructs that are shown in a JSP data structure diagram.

1 .....

.....

2 .....

.....

[2]

5 A declarative programming language is used to represent the following knowledge base.

```
01 person(william).
02 person(deeraj).
03 person(ingrid).
04 person(meghan).
05 country(england).
06 country(spain).
07 country(bangladesh).
08 country(new_zealand).
09 country(malaysia).
10 country(mauritius).
11 visited(william, spain).
12 visited(ingrid, new_zealand).
13 visited(deeraj, spain).
14 visited(meghan, spain).
```

These clauses have the following meanings:

Clause	Meaning
02	Deeraj is a person
05	England is a country
11	William has visited Spain

(a) Gina is a person who has visited Cyprus.

Write additional clauses to represent this information.

15 .....

16 .....

17 .....

[3]

(b) Write the result returned by the goal:

```
visited(X, spain).
```

X = ... [2]

(c) P might visit C, if P is a person, C is a country and P has not visited C.

Write this as a rule.

```
mightvisit(P , C)
```

IF .....

.....

..... [4]

6 Object-oriented programming has several features. These include containment, classes, methods and properties.

(a) Describe what is meant by **containment**.

.....

.....

.....

.....

.....

..... [3]

(b) Identify **two other** features of object-oriented programming.

1 .....

2 .....
[2]

7 A programmer is creating a computer game. The programmer has designed the class, `Character`, for the characters in the game.

The following class diagram shows the design for the `Character` class.

Character		
`Name : STRING`	`//`	`initialised in constructor to the parameter value passed to the constructor`
`Skill : INTEGER`	`//`	`initialised in constructor to 0`
`Health : INTEGER`	`//`	`initialised in constructor to 50`
`Shield : INTEGER`	`//`	`initialised in constructor to a random value between 1 and 25 (inclusive)`
`Constructor()`	`//`	`method used to create and initialise an object`
`GetName()`	`//`	`returns Name value`
`GetSkill()`	`//`	`returns Skill value`
`GetHealth()`	`//`	`returns Health value`
`GetShield()`	`//`	`returns Shield value`
`SetSkill()`	`//`	`increases Skill by the parameter value`
`SetHealth()`	`//`	`increases or decreases Health by the parameter value`
`SetShield()`	`//`	`increases or decreases Shield value by the parameter value`

(a) Write **program code** for the `Constructor()` method. Use the appropriate constructor method for your chosen programming language.

Programming language .....

Program code

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

(b) Write **program code** for the `GetSkill()` method.

Programming language .....

Program code

.....

.....

.....

.....

..... [2]

(c) The method `SetSkill()` validates the parameter value and updates the value of `Skill`.

The method is passed an `INTEGER` parameter that must be between 10 and 25 (inclusive). A value outside of this is not valid.

If the parameter value is valid, the method will increase `Skill` by the parameter value. The maximum value that `Skill` can be increased to is 200. For example:

- `Skill` currently stores 180
- it is passed a valid parameter value of 25
- `Skill` will now store 200.

The method must return:

- `-1` if the parameter value is not valid
- `1` if the value of `Skill` is updated **and** `Skill` is less than 200
- `0` if the value of `Skill` is `200`.

Write **program code** for the `SetSkill()` method.

Programming language .....

Program code

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]

(d) There are five characters in the game. All the character objects are stored in a 1D array.

Write **pseudocode** to declare the array, `CharacterArray`, to store the five character objects.

.....

..... [2]

(e) The game has the character with the name Victory.

Write **program code** to create the character Victory as an instance of the class `Character`. The object needs to be stored in the first element of the array `CharacterArray`.

Programming language .....

Program code

.....

.....

..... [3]

Question 8 begins on the next page.

8 Files can be structured in serial, sequential or random format.

Tick (✓) **one** box in each row to show whether the statement applies to **Serial**, **Sequential** or **Random** format.

Statement	Serial	Sequential	Random
Uses a hashing algorithm			
No key field is used when storing data, for example, it is stored in chronological order			
Collisions can occur			
Least efficient for a very large number of records			
Most efficient for a very large number of records			

[3]

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and cleared where possible. Every reasonable effort has been made by the publisher (UCLES) to trace copyright holders, but if any items requiring clearance have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced online in the Cambridge Assessment International Education Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cambridgeinternational.org after the live examination series.

Cambridge Assessment International Education is part of the Cambridge Assessment Group. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which itself is a department of the University of Cambridge.

© UCLES 2020