Cambridge Assessment International Education

Cambridge International AS & A Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

8888903826 4

COMPUTER SCIENCE **9608/41**

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 Carlos is writing exception handling code for his program.

(a) State what is meant by an **exception**.

.....

..... [1]

(b) Give **three** situations where an exception handling routine would be required.

1 .....

.....

2 .....

.....

3 .....

.....

[3]

(c) Describe the benefits of using exception handling in a program.

.....

.....

.....

..... [2]

2 **(a)** Programs can be written using recursion.

Tick (✓) one or more boxes to show the features that **must** be included in a valid recursive algorithm.

Feature	Must be included
Incrementation	
General case	
Base case	
Selection case	
It calls itself	

[2]

(b) The following recursive procedure outputs every even number from the positive parameter value down to and including 2.

The procedure checks if the integer parameter is an even or an odd number. If the number is odd, the procedure converts it to an even number by subtracting 1 from it.

The function `MOD(ThisNum : INTEGER, ThisDiv : INTEGER)` returns the remainder value when `ThisNum` is divided by `ThisDiv`.

Complete the **pseudocode** for the recursive procedure.

```
PROCEDURE Count(BYVALUE ..... : INTEGER)
    IF .....(Number, 2) <> 0
        THEN
            Number ← Number - 1
    ENDIF
    OUTPUT .....
    IF Number > 0
        THEN
            .....(..... - 1)
    ENDIF
ENDPROCEDURE
```

[5]

(c) A program allows guests to input a meal option at a wedding.

Guests can choose meal option 1 or meal option 2.

The program will keep count of the numbers of each meal option chosen.

The program ends when a value other than 1 or 2 is entered. It then outputs the count of each meal option.

```
PROCEDURE MealsCount(BYREF MealOption1 : INTEGER, MealOption2 : INTEGER)

    DECLARE MealOption : INTEGER

    DECLARE MoreMeals : BOOLEAN

    MoreMeals ← True

    WHILE MoreMeals = True

        INPUT MealOption

        IF MealOption = 1

            THEN

                MealOption1 ← MealOption1 + 1

            ELSE

                IF MealOption = 2

                    THEN

                        MealOption2 ← MealOption2 + 1

                    ELSE

                        OUTPUT MealOption1, " ", MealOption2

                        MoreMeals ← False

                ENDIF

        ENDIF

    ENDWHILE

ENDPROCEDURE
```

The program contains a conditional loop.

Use **pseudocode** to rewrite the conditional loop as a recursive algorithm.

```
PROCEDURE MealsCount(BYREF MealOption1 : INTEGER, MealOption2 : INTEGER)
    DECLARE MealOption : INTEGER
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

```
ENDPROCEDURE
```

[5]

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

```
01 person(jessica).
02 person(pradeep).
03 person(steffi).
04 person(johann).
05 sport(football).
06 sport(hockey).
07 sport(cricket).
08 sport(volleyball).
09 plays(johann, football).
10 plays(steffi, cricket).
11 plays(jessica, football).
12 will_not_play(pradeep, cricket).
```

These clauses have the following meanings:

Clause	Meaning
01	Jessica is a person
05	Football is a sport
09	Johann plays football
12	Pradeep refuses to play cricket

(a) Elle is a person who plays rugby but refuses to play hockey.

Write additional clauses to represent this information.

13 .....

14 .....

15 .....

16 .....

[4]

(b) Write the result returned by the goal:

```
plays(X, football).
```

X = ... [1]

(c) Y might play X, if Y is a person, X is a sport and Y does not refuse to play X.

Write this as a rule.

```
mightplay(Y , X)
```

IF .....

.....

..... [5]

4 Object-oriented programming has several features. These include inheritance, classes, methods and properties.

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

.....

.....

.....

..... [2]

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

1 .....

2 .....
[2]

5 A tennis club is developing a program to store details of the lessons it offers. The programmer has designed the class `Lesson` for the details of the lessons.

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

Lesson

`LessonType : STRING // initialised in constructor to the parameter`
`// value passed to the constructor`
`Instructor : STRING // initialised in constructor to the parameter`
`// value passed to the constructor`

`Constructor() // method used to create and initialise an`
`// object`
`GetLessonType() // returns LessonType value`
`GetInstructor() // returns Instructor value`
`GetFee() // returns the cost of a lesson`
`SetLessonType() // sets the LessonType to the parameter value`
`SetInstructor() // sets the Instructor to 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

.....

.....

.....

.....

.....

.....

.....

..... [3]

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

Programming language .....

Program code

.....

.....

.....

.....

.....

..... [2]

(c) `Fee` is the cost that a customer will pay for a lesson.

The method `GetFee()` validates the parameter value. The method is sent a parameter value that represents the skill level of the customer: beginner, intermediate or advanced.

The parameter will be a character:

- `'B'` for beginner
- `'I'` for intermediate
- `'A'` for advanced.

The method must check the parameter value is a valid character (`'B'`, `'I'` or `'A'`) and return the correct fee. It must return `-1` if it is not a valid character.

The fees are:

- $45 for a beginner
- $50 for an intermediate
- $55 for an advanced.

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

Programming language .....

Program code

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

(d) The tennis club only offers nine different types of lesson. The lesson objects are stored in a 1D array.

Write **pseudocode** to declare an array `LessonArray` to store the nine lesson objects.

.....

..... [2]

(e) The tennis club has the lesson 'Improve Your Serve' that has David as the instructor.

Write **program code** to create the lesson 'Improve Your Serve' as an instance of the class `Lesson`. The object needs to be stored in the third element of the array `LessonArray`.

Programming language .....

Program code

.....

.....

..... [3]

6 A theatre company stores customer login details to allow customers to book tickets online.

A hash table stores login details for 2000 customers.

Each customer's details are stored in a record.

The declaration for `CustomerRecord` is:

```
TYPE CustomerRecord

      DECLARE UserID : STRING

      DECLARE PINNumber : INTEGER

ENDTYPE
```

A 1D array, `CustomerDetails`, is used to implement the hash table. `CustomerDetails` is a global array. The 1D array has 6000 elements.

(a) The procedure `InitialiseHashTable()` initialises the hash table. `UserID` is initialised as an empty string, and `PINNumber` initialised to `0` for all of the records.

Write **pseudocode** for the procedure `InitialiseHashTable()`.

.....

.....

.....

.....

.....

.....

.....

..... [4]

(b) The function `InsertRecord()` is used to insert a new record into the hash table.

The function, `Hash()`:

- takes a `UserID` as a parameter
- performs the hashing algorithm
- returns the calculated index of the user ID within the hash table.

If the hash table is full, the function `InsertRecord()` returns `-1`. If there is space available in the hash table, the record is inserted, and it returns the position of this record in the array.

© UCLES 2020

Complete the **pseudocode** for the function.

```
FUNCTION InsertRecord(NewRecord : CustomerRecord) RETURNS INTEGER
    DECLARE Count : INTEGER
    DECLARE Index : INTEGER
    Count ← 0
    Index ← Hash(.....)
    WHILE (CustomerDetails[Index].UserID <> "") ..... (Count <= 5999)
        Index ← Index + 1
        Count ← Count + 1
        IF Index > 5999
            THEN
                .....
        ENDIF
    ENDWHILE
    IF Count > 5999
        THEN
            .....
        ELSE
            CustomerDetails[.....] ← .....
            .....
    ENDIF
ENDFUNCTION
```

[7]

7 **(a)** A shirt design company has an order form to order shirts. Customers can order multiple shirts using the same form.

The customer details section has the data:

- name
- address
- telephone number.

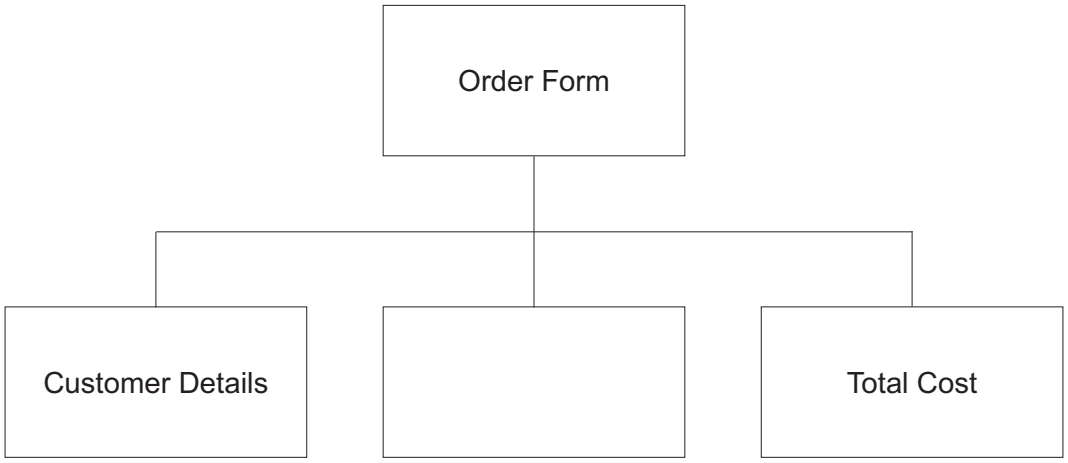
The order details section has the data:

- shirt ID
- colour
- cost.

A total cost for the order is also calculated.

The cost of each shirt is dependent on the size ordered. The sizes customers can order are small, medium and large.

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



[7]

(b) Each customer's order is stored as a record in a file. The customers' orders are stored in the order in which they arrive in the file and no key field is used.

(i) Identify this type of file structure.

..... [1]

(ii) Identify **two other** types of file structure.

1 .....

2 .....
[2]

(c) The procedure `UpdateTelephone()` allows the shirt company to update the record for a customer's details. The procedure will update the telephone number.

The program stores customer details as a custom data type, `Customer`.

The definition for this data type is:

```
TYPE Customer

    Name : STRING

    Address : STRING

    TelephoneNumber : STRING

ENDTYPE
```

The procedure `UpdateTelephone()` takes the customer record to be updated and the new telephone number as parameters. It then updates the telephone number in the record.

Complete the **pseudocode** for the procedure `UpdateTelephone()`.

```
PROCEDURE UpdateTelephone(..... ThisCustomer : Customer,

          ..... NewTelephoneNumber : STRING)

    .....

ENDPROCEDURE
```
[3]

(d) The shirt company is looking to implement a system to reward customers. The system includes:

- 10% discount on orders over $50
- free gift if order over $50 and if the order is placed on a Monday
- additional 5% discount if a customer has a loyalty card
- free delivery for a customer with a loyalty card and spends over $50.

Complete the following decision table for this system.

Conditions	**Order over $50**	**Y**	**Y**	**Y**	**Y**	**N**	**N**	**N**	**N**
	Monday	**Y**	**Y**	**N**	**N**	**Y**	**Y**	**N**	**N**
	Loyalty card	**Y**	**N**	**Y**	**N**	**Y**	**N**	**Y**	**N**
Actions	**Additional 5% discount**								
	10% discount								
	Free gift								
	Free delivery								

[4]

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