www.dynamicpapers.com

Cambridge Assessment International Education

Cambridge International AS & A Level

COMPUTER SCIENCE **9608/42**

Paper 4 Written Paper **May/June 2020**

MARK SCHEME

Maximum Mark: 75

Published

Students did not sit exam papers in the June 2020 series due to the Covid-19 global pandemic.

This mark scheme is published to support teachers and students and should be read together with the question paper. It shows the requirements of the exam. The answer column of the mark scheme shows the proposed basis on which Examiners would award marks for this exam. Where appropriate, this column also provides the most likely acceptable alternative responses expected from students. Examiners usually review the mark scheme after they have seen student responses and update the mark scheme if appropriate. In the June series, Examiners were unable to consider the acceptability of alternative responses, as there were no student responses to consider.

Mark schemes should usually be read together with the Principal Examiner Report for Teachers. However, because students did not sit exam papers, there is no Principal Examiner Report for Teachers for the June 2020 series.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the June 2020 series for most Cambridge IGCSE™ and Cambridge International A & AS Level components, and some Cambridge O Level components.

© UCLES 2020

www.dynamicpapers.com

Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptors for a question. Each question paper and mark scheme will also comply with these marking principles.

GENERIC MARKING PRINCIPLE 1:

Marks must be awarded in line with:

- the specific content of the mark scheme or the generic level descriptors for the question
- the specific skills defined in the mark scheme or in the generic level descriptors for the question
- the standard of response required by a candidate as exemplified by the standardisation scripts.

GENERIC MARKING PRINCIPLE 2:

Marks awarded are always **whole marks** (not half marks, or other fractions).

GENERIC MARKING PRINCIPLE 3:

Marks must be awarded **positively**:

- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can do
- marks are not deducted for errors
- marks are not deducted for omissions
- answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous.

GENERIC MARKING PRINCIPLE 4:

Rules must be applied consistently e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

www.dynamicpapers.com

GENERIC MARKING PRINCIPLE 5: Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen).
GENERIC MARKING PRINCIPLE 6: Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

www.dynamicpapers.com

Question	Answer	Marks
1(a)	**1 mark** per reason to **max 3** e.g. • Division by zero • Array out of bounds • File does not exist • Stack overflow • Memory leakage • Hardware fault/failure	**3**

1(b)	**1 mark** per bullet point
	
	• Check if file exist
	• Reporting appropriate exception
	
	Python
	`try:`
	`   file = open('MyData.txt')`
	`except:`
	`   print "No file found"`
	
	Visual Basic (.net)
	`Try`
	`  Dim fileReader As New System.IO.StreamReader("MyData.txt")`
	`Catch ex As Exception`
	`  console.writeline("No file found")`
	`End Try`
	
	Pascal
	`try`
	`  Readln("MyData.txt")`
	`except`
	`  Writeln("No file found")`
	`end;`
		2

www.dynamicpapers.com

Question	Answer	Marks

2(a)	**1 mark** per bullet point to **max 4**
	
	• Declaration of array with 3000 elements
	• … of type `CustomerRecord`
	• Looping 3000 times
	• Accessing the username and password for each field
	• Correct initialisation of values to “”
	
	`DECLARE CustomerLogIn : ARRAY[0:2999] OF CustomerRecord`
	`FOR x ← 0 TO 2999`
	`  CustomerLogIn[x].Username ← ""`
	`  CustomerLogIn[x].Password ← ""`
	`ENDFOR`
		4

www.dynamicpapers.com

Question	Answer	Marks
2(b)(i)	**1 mark** for each correctly filled in space	**7**

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

Question	Answer	Marks
2(b)(ii)	**1 mark** per bullet point to **max 2** • Records if you have checked every record // the number of records checked in the array • … without finding the search value • … and stops infinite loop // stopping condition	**2**

www.dynamicpapers.com

Question	Answer	Marks
3(a)	1 mark per bullet point • Have a base case // stopping condition • Work toward the base case // general case // changes state • Call itself // defined in terms of itself	**3**

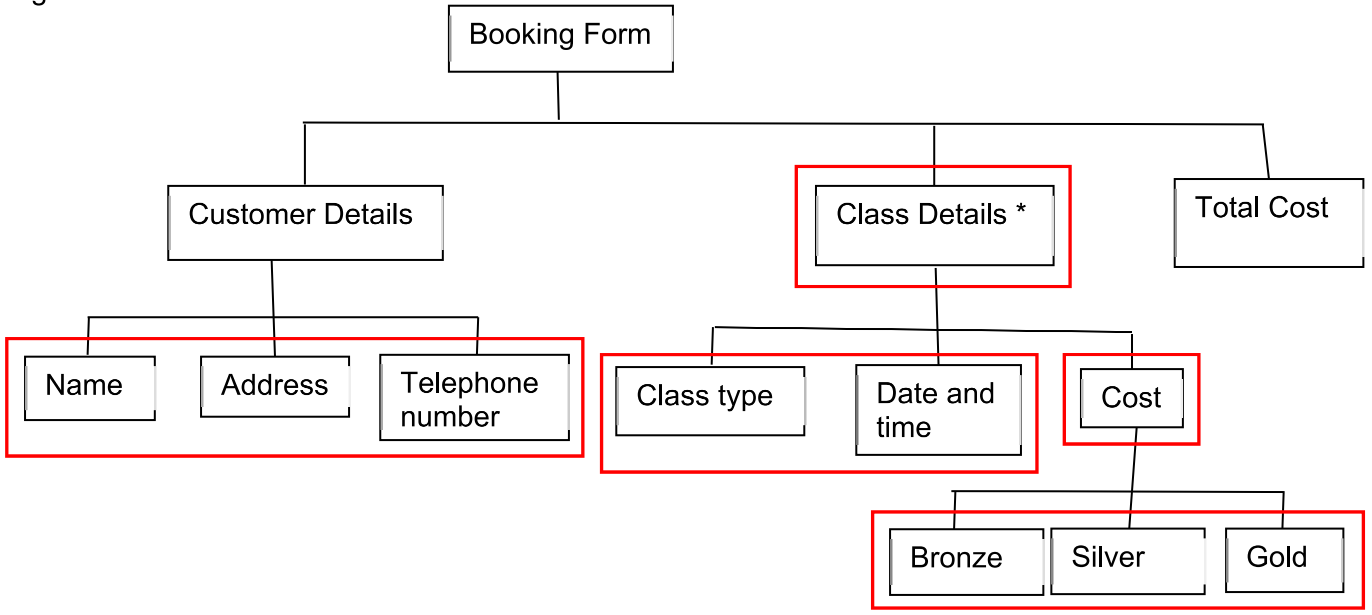
3(b)	1 mark for each (shown in **bold**)
	
	`FUNCTION IsPalindrome(CheckWord : STRING) RETURNS BOOLEAN`
	`    IF` **`Length(CheckWord)`** `<= 1`
	`        THEN`
	`            RETURN` **`True`**
	`    ENDIF`
	`    IF Substring(CheckWord, 0, 1) <>`
	`              Substring(CheckWord,`**`Length`**`(CheckWord)-1, 1)`
	`        THEN`
	`            RETURN` **`False`**
	`        ELSE`
	`            RETURN` **`IsPalindrome`**`(Substring(CheckWord,1,`
	`                                Length(CheckWord)-2))`
	`    ENDIF`
	`ENDFUNCTION`
		5

www.dynamicpapers.com

Question	Answer	Marks

3(c)	**1 mark** per bullet point
	
	• Function is defined and returns integer with correct parameters
	• Returns 1 if exponent is 0 // returns base if exponent is 1
	• Calculates base to power with recursive call …
	• … with correct parameters
	• Returns result of calculation
	e.g.
	`FUNCTION FindPower(Base:INTEGER, Exp: INTEGER) RETURNS INTEGER`
	`    DECLARE Value : INTEGER`
	`    IF Exp = 0`
	`        THEN`
	`            RETURN 1`
	`        ELSE`
	`            Value ← Base * FindPower(Base, Exp - 1)`
	`            RETURN Value`
	`    ENDIF`
	`ENDFUNCTION`
		5

www.dynamicpapers.com

Question	Answer	Marks
4(a)	**1 mark** per bullet point: • Name, Address and Telephone number beneath Customer details • Class Details (or equivalent) below booking form • … with appropriate iteration of entering Class Details • Class type, and Date and time beneath Class Details (or equivalent) • Cost beneath Class Details (or equivalent) … • … with Bronze, Silver and Gold below it (FT for name and position of Cost) … • … with selection (FT for name and position of B, S, G) e.g.  Booking Form Customer Details Class Details * Total Cost Name Address Telephone number Class type Date and time Cost Bronze Silver Gold	7
4(b)	**1 mark** for each feature to **max 2** • Sequence • Selection • Iteration	2

www.dynamicpapers.com

Question	Answer	Marks
5(a)	• person(gina) • country(cyprus) • visited(gina, cyprus)	**3**

5(b)	**1 mark** for 2 correct, **2 marks** for 3 correct
	
	`william`
	`deeraj`
	`meghan`
		2

5(c)	**1 mark** per bullet point
	• `person(P) // country(C)`
	• `AND country(C) // AND person(P)`
	• `AND NOT // , NOT`
	• `visited(P, C)`
	
	`mightvisit(P, C)`
	`IF person (P) AND country (C) AND NOT visited(P, C)`
		4

Question	Answer	Marks
6(a)	**1 mark** per bullet point to **max 3** • A class includes an instance of another class • Aggregation • … the contained object can exist outside of its super class • Composition • … object is declared and exists within its super class // object is destroyed when super class is destroyed	**3**
6(b)	**1 mark** per feature to **max 2** e.g. • Inheritance • Polymorphism • Encapsulation • Private/public	**2**

www.dynamicpapers.com

Question	Answer	Marks

7(a)	**1 mark** per bullet point to **max 5**
	• Method header and close (where applicable) …
	• … with correct parameter
	• Initialised Name to parameter value
	• Initialised Skill to 0 and Health to 50
	• Initialised Shield to randomly generated value
	• … between 1 and 25
	
	Python
	`def __init__(self, CharacterName) :`
	`    self.__Name = CharacterName`
	`    self.__Skill = 0`
	`    self.__Health = 50`
	`    self.__Shield = random.randint(1, 25)`
	
	Pascal
	`Constructor Character.Create(CharacterName);`
	`begin`
	`  Name := CharacterName;`
	`  Skill: = 0;`
	`  Health := 50;`
	`  Shield: = random(1,25);`
	`end;`
	
	Visual Basic (.net)
	`Public Sub New(CharacterName)`
	`  Name = CharacterName`
	`  Skill = 0`
	`  Health = 50`
	`  Shield =`
	`End Sub`
		5

www.dynamicpapers.com

Question	Answer	Marks

7(b)	**1 mark** per bullet point
	• function header and close (where applicable)
	• returns Skill
	e.g.
	
	Python
	
	`def GetSkill(self) :`
	`    return self.__Skill`
	
	Visual Basic (.net)
	`public function GetSkill() AS Integer`
	`  return(Skill)`
		2

7(c)	**1 mark** per bullet to **max 6**
	• Function header and close (where applicable)
	• …Takes a parameter value
	• Check parameter is >= 10 and <=25 and returns -2
	• Checks new Value of skill is not > 200
	• …limits to 200 if over
	• …returns 0
	• Updates skill value and returns 1
	e.g.
	
	Python
	
	`def SetSkill(self, Value) :`
	`    if Value < 10 or Value > 25 :`
	`        return -1`
	`    else :`
	`        if self.__Skill + Value >= 200:`
	`            Skill = 200`
	`            return 0`
	`        else :`
	`            self.__Skill = self.__Skill + Value`
	`            return 1`
		6

www.dynamicpapers.com

Question	Answer	Marks
7(d)	• Array declaration of correct name and size • Correct data type used `DECLARE CharacterArray : ARRAY[0:4] OF Character`	**2**

7(e)	**1 mark** per bullet point
	• Object is created using constructor
	• "Victory" passed as parameter
	• Stored in correct index of array
	
	PYTHON
	`CharacterArray[0] = Character("Victory")`
	
	Visual Basic (.net)
	`CharacterArray[0] = New Character("Victory")`
		3

Question	Answer	Marks
8	See table below	**3**

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