ADVANCED SUBSIDIARY GCE   **2507**

COMPUTING

Structured Practical Computing Tasks

MARK SCHEME

JUNE 2008   Tasks issued to candidates September 2007

Maximum mark **120**

INSTRUCTIONS TO TEACHERS

- This mark scheme must not be opened until all the work of all candidates at the Centre has been received, ready for final marking.

- Once this mark scheme has been opened, candidates must have no further access to their work until 30 June 2008.

This document consists of **10** printed pages and **2** blank pages.

Task 1 [39 marks]

(a) **Student (10 marks):** *Suitable descriptive attribute names, reasons and data types must all be correct to obtain the marks. Reasons for having particular attributes must refer to why they are needed in this particular database.*

*Give **1 mark** for each of the following mark points:*
- Student number/ID – to have a unique reference to each student **and** to be used as the key.
  Type – text/string (not numeric/integer).
- Name – to identify the name of the student for reports/contact.
  Type – text/character/string.
- Address – to be able to send results to student.
  Type – text/character/string/alphanumeric.
- Post code – to be able to send results to student.
  Type – text/character/string/alphanumeric.
- Phone number – for quick contact.
  Type – text/character/string/alphanumeric.
- Email – for quick contact.
  Type – text/character/string/alphanumeric.
- Gender – to be able to separate results according to gender for statistical purposes.
  Type – text/character/string/alphanumeric.
- DOB – to be able to separate results according to age for statistical purposes.
  Type – date.

*Give **1 mark** for each of the following to a **maximum of 2 marks**. Award this mark only where the candidate has clearly explained their validation process.*
- Validation of student number/ID that ensures that each ID is unique.
- Validation of student number/ID that ensures that each ID consists of exactly 8 digits.
- Validation of DOB that ensures that only reasonable dates have been entered.
- Validation of gender to ensure that only one of two possibilities is entered for each student.

[10]

(b) Module (4 marks): S*uitable descriptive attribute names, reasons and data types must all be correct to obtain the marks. Reasons for having particular attributes must refer to why they are needed in this particular database.*

*Give **1 mark** for each of the following mark points:*
- Module number/ID – to have a unique reference to each module **and** to be used as the key.
  Type – alphanumeric/text/string/numeric/integer/number.
- Name – To give a sensible name to the module.
  Type – alphanumeric/text/string.
- Credits – To be able to know how many credits a student has received.
  Type – numeric/integer/number.
- Validation of credits so that only 15 or 30 can be entered.

[4]

(c) Result (7 marks): *Suitable descriptive attribute names, reasons and data types must all be correct to obtain the marks. Reasons for having particular attributes must refer to why they are needed in this particular database.*

*Give **1 mark** for each mark point*
- Student Number/ID – to link to STUDENT table.
  Type – same as in STUDENT table.
- Module Number/ID – to link to MODULE table.
  Type – same as in MODULE table.
- Month (of Examination) – to know which session the examination was in.
  Type – alphanumeric/text/string.
- Year (of Examination) – to know which year the examination was in.
  Type – alphanumeric/text/string/numeric/integer.
- Result – to know how well the student did.
  Type – integer/numeric/number.
- The composite key is the first four attributes.

*Give **1 mark** for each mark point to a maximum of 1*
- Validation of month – can only be Jan or Jun.
- Validation of year – must not be greater than current year.

[7]

(d) Tables (8 marks): *To gain a mark point the fields shown in a table must match and include all those defined by the candidate in (a), (b) and (c).*

*Give **1 mark** for each of the following mark points:*
- The STUDENT table has at least 20 records **and** every Student ID is different.
- The MODULE table has at least 10 records **and** every Module ID is different.
- The RESULT table has at least 30 entries.

*Give **1 mark** for each of the following mark points, to a **maximum of 5**.*
- At least two students have taken more than one module in the same session.
- At least two students have taken modules in more than one session.
- At least two students have taken a module more than once.
- At least one student has not taken a module.
- At least two modules have been taken by more than one student.
- At least one module has been taken in only one session.
- At least one module has never been taken.

[8]

(e) Report (5 marks)

*Give **1 mark** for each of the following mark points:*
- The report has a suitable heading.
- The report has a date.
- The report contains the data of all the modules taken by a student.
- The report is grouped on student name and the name only occurs once (not for each module a student has taken).
- The report is ordered on student name.

[5]

(f) Report (5 marks)

*Give **1 mark** for each of the following mark points:*
- The report heading includes the module name and/or ID.
- The report heading includes the month.
- The report heading includes the year.
- The report has a date.
- Results are ordered by student name.

[5]

Task 2 [32 marks]

(a) cd (6 marks)

*Give **1 mark** for each row:*

current	anArray	newString	rank
	(		0
c	(c		
d	(	c	1
	(d		
)	(	cd	2

Output is: Invalid string

[6]

(b) p*(q (9 marks)

*Give **1 mark** for each row:*

current	anArray	newString	rank
	(		0
p	(p		
*	(	p	1
	(*		
(	(*(		
q	(*(q		
)	(*(	pq	2
	(*		

Output is: Invalid string

[9]

(c) (v-w)*(x+y) (17 marks)

*Give **1 mark** for each row except the first three:*

current	anArray	newString	rank
	(		0
(	((		
v	((v		
-	((	v	1
	((-		
w	((-w		
)	((-	vw	2
	((	vw-	1
	(		
*	(*		
(	(*(		
x	(*(x		
+	(*(	vw-x	2
	(*(+		
y	(*(+y		
)	(*(+	vw-xy	3
	(*(	vw-xy+	2
	(*		
)	(	vw-xy+*	1

Output: Valid string

[17]

Task 3 [32 marks]

(a) Initialisation – 6 marks

*Give **1 mark** for each of the following mark points:*
- The code is fully annotated.
  It would be possible to create suitable code with only the annotation as guidance (this condition applies also to later sections).
- The data structures allow the program to store the number of marbles in each small cup.
- The data structures allow the program to store the number of marbles in each large cup.
- The data structures have meaningful names.
  For variables, labels, buttons, etc (this applies also to later sections).
- The contents of each small cup is set to 4.
- The contents of each large cup is set to 0.

[6]

(b) Interface – 5 marks

*Give **1 mark** for each of the following mark points:*
- The code is fully annotated.
- Meaningful names are used throughout.
- The interface shows all the cups, small and large.
- The interface shows the number of marbles in each small cup.
- The interface shows the number of marbles in each large cup.

[5]

(c) Start – 4 marks

*Give **1 mark** for each of the following mark points:*
- The code is fully annotated.
- Meaningful names are used throughout.
- The input only allows A or B to be entered
- or allows the players to quit.
  Evidence of testing is required for the last two marks.

[4]

(d) Move – 5 marks

*Give **1 mark** for each of the following mark points to a maximum of 5:*
- The code is fully annotated
- The input allows the player to surrender.
- The cup number can only be an integer
- in the range 1 to 6 for player A
- or 7 to 12 for player B.
- The code rejects an empty cup.
  Evidence of testing is required for the last four marks.

[5]

(e) Distribution – 5 marks

*Give **1 mark** for each of the following mark points:*
- The code is fully annotated.
- The marbles are correctly distributed.
- The code correctly empties any cup(s) if necessary.
- Captured marbles are placed in the correct large cup.
- The results are correctly displayed.
  Evidence of testing is required for the last three marks.

[5]

(f) The game – 7 marks

*Give **1 mark** for each of the following mark points:*
- The code is fully annotated.
- The play alternates between players.
- The game continues until one player wins
- or a player surrenders
- or no player can win
- and the remaining marbles are placed correctly in the large cups.
- The result is reported.
  Evidence of testing is required for the last six marks.

[7]

Task 4 [17 marks]

(a) Output (7 marks)

*Give **1 mark** for each row **plus 1** if they are in the right order:*
RED
RDE
ERD
EDR
DRE
DER

[7]

(b) Trace (10 marks)

*Give **1 mark** for each line after the first:*
MYSTERY("", "AB")
length = 2
i = 1
    MYSTERY("A", "B")
    length = 1
    PRINT "AB"
i = 2
    MYSTERY("B", "A")
    length = 1
    PRINT "BA"
END of procedure

[10]

10

BLANK PAGE

BLANK PAGE