



CENTRE REGIONAL MOCK EXAMINATION IN PARTNERSHIP WITH CASPA	SUBJECT CODE NUMBER 795	PAPER NUMBER 3
	SUBJECT TITLE COMPUTER SCIENCE	
CANDIDATE NAME: CANDIDATE NUMBER: CENTRE NUMBER:		
ADVANCED LEVEL	DATE: To be scheduled by the school	

PRACTICAL

Time Allowed: TWO hours

Carry out ALL the task given. For your guidance, the appropriate marks for each part of a task is indicate in brackets.

Great importance is attached to the accuracy, layout and labeling of the diagrams and computer generated outputs.

You are reminded of the necessity for good English and orderly presentation in your answers.

Analyse the problem as well as write algorithms in the answer booklet provided. Also record in your answer booklet ant information requested or that you believe would make it easier to understand how you carried out tasks or answered questions.

You are expected to print a single copy of relevant fragments of your work at different times. Please notify the supervisor of any required printout that was not done!

When an imperative programming language is required to write program code, ether standard (ISO) Pascal or the standard (ANSI) C programming language may be used.

If need be, supervisors will assist you in recording details of intermediate work carried out on the computer.

Where information is provided is provided as soft copy, notify the supervisor if it is not found in your machine or has not been made available to you.

The term ATM stands for automated teller machine. It is an electronic device that is used by only bank customers to process account transactions. The users access their accounts through a special type of plastic card that is encoded with user information on a magnetic strip.

Below are some of the transactions or operations found in an ATM:

1. Withdrawals are the most common way to use an ATM. You simply get cash out from your account. For a withdrawal, you'll just indicate how much you want to take out (usually in increments of 150FCFA for transaction fee).
2. **Balance Inquiry** show you how much money you have. Selecting this option will display your current account balance.

In case you have chosen to do a withdrawal, you will be prompt to enter an amount. In case the amount entered is superior to the amount found in your account, the transaction will fail. Otherwise, you will be prompted to know if you need a Printed receipt; which you choose to print or not.

You have been contacted to implement this automated teller machine in a programming language of your choice. After thorough explanation on how the ATM functions, you have been provided with A sample of the whole program called *sample.c*, use this sample code to implement the following tasks.

Task 1

1. State **ONE** input and output device used in this system. (2 marks)
 2. During the changing of the 4-digit PIN, which verification check is used? (1 mark)
 3. State and explain **ONE** validation check used in this system. (2 marks)
-

Task 2

Once the user inserts the ATM card into the card reader, a message appears on the screen prompting for the user to choose a language of operation. Once the language is selected, the screen is cleared and user is prompt to enter a PIN which is to be verified from the information found in the **Account** array. All the pseudocode functions and procedures you will need to write the program are found in the **Appendix**. All variables are to be initialize and declared as global variables.

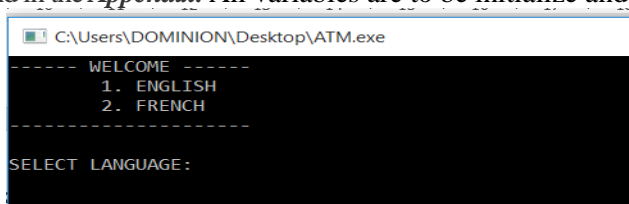


Fig 1: Language Menu

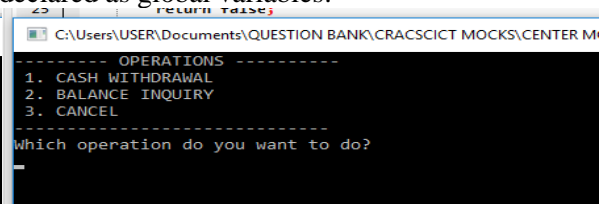
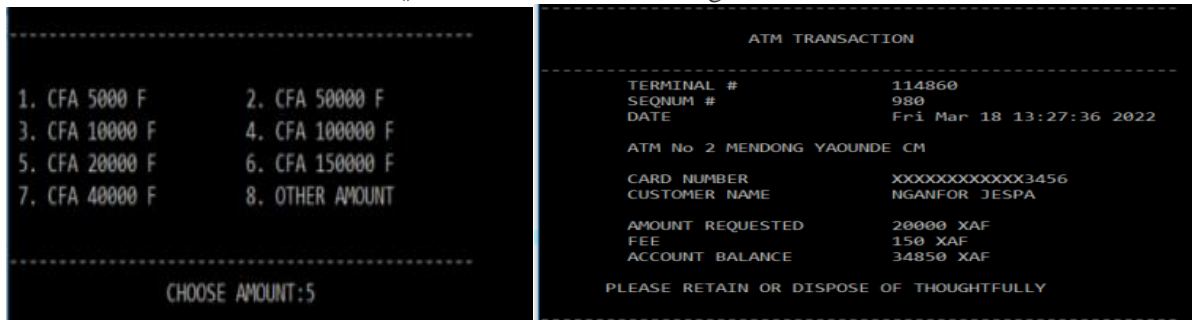


Fig 2: Display Menu

4. Write the above menu in a programming language of your choice. Make sure you type the code in the main function. (2marks)
 5. Write the procedures *English()* and *French ()* in a programming language. French () is the same as English, just the outputs are been translated for the French user to understand. (5 marks)
 6. Write a procedure *Display ()* that displays a menu in French or English depending on the language chosen. Save your program as *Task 2.c*. Run the program. Screenshot the output and Save it as *Task 2a*. (5marks)
-

Task 3

7. Write a Procedure *withdrawal* () which does the following:



- Ask the user to choose an amount or enter an amount if not found in the list.
- If the amount entered is greater than the user's balance, then a message is sent "Insufficient funds" and display () called. Otherwise, the amount is being deducted from the user's account balance.
- The user is being asked if he/she needed a receipt. If yes, a receipt is printed as above.
- Otherwise, a message "THANKS FOR YOUR VISIT" is sent.

(4 marks)

8. Write the procedure *PrintBalance* () and *GetBalance* () in a programming language.

(5 marks)

9. Complete the rest of the program, following the comments written in the code. Save your program as *Task 3.c*. Run the Program, screenshot the output for a cash withdrawal of 5000FCFA and Save as *Task 3a*.

(4 marks)

NB: Print all the screen captures and the source codes.

DATABASE: SCHOOL CLUBS' SYSTEM

(30 marks)

An after-school club runs different courses and has specialist teachers teaching one course only. Whereas one student can choose to attend various courses.

Details of the Student, Course, Teacher and CourseEnrolled are stored in a relational database using the following four relations:

Student (*StudentId*, *ForeName*, *SurName*, *DOB*, *Address*, *City*)

Course (*CourseId*, *Title*, *Duration*, *Fee*, *CourseType*)

CourseEnrolled (*StudentId*, *CourseId*)

Teacher (*TeacherId*, *ForeName*, *SurName*, *CourseId*)

In this system, the following restrictions apply to some attributes:

- *CourseId*: a mixture of **exactly** 6 letters and numbers, e.g., Step01
 - Same rule applies to other IDs.
- *Duration*: a whole number representing the length of the course e.g., 33
- *CourseType*: can be either 'Weekly' or 'Monthly' and nothing else
- *Fee*: a monetary value, eg 100

Task 4 – Designing the Database

10. Write an SQL statement to create a database named **CLUB** using any suitable software package. (1 mark)
 11. Create the table **Student, Course, Teacher, and CourseEnrolled**.
Write the SQL statement to create the table **Course** in your answer booklet. (3 marks)
 12. Create a relationship diagram between all the tables.
Screenshot the relationships in the database and save it as **Relationships**. (2 marks)
 13. Do three records for each of the relations. (8 marks)
-

Task 5 – Queries

14. The school has introduced another course for GCE-Computer-FT which is fast track i.e., can be done in one year with a fee of 5000FCFA and has **CourseId** GCEC02.
Write the SQL statement to add this new course into Course table. Save the query as **New**. (3 marks)

CourseID: GCEC02 Title: GCE-Computer-FT Duration: 12
Fee: 5000 Course type: Monthly
15. After above change, now the course with name GCE-Computer-FT has been changed to GCE-Computer-ST.
Write the SQL statement to update the data in the Course table to reflect this change.
Save query as **Update**. (3 marks)
16. The school wants to have a list of students with their forename and surname who attend weekly courses taught by a teacher William Smith. The result should be in ascending order of students' forenames.
Write an SQL query to achieve this. (4 marks)
17. The school has now decided to incorporate parents' database into this structure. Most of the children have their siblings enrolled in the school as well. The database must keep a record of all the parents who have their children enrolled in the school.
 - a) Explain how you would change the design of the database so that the information about parents can be stored. (3 marks)
 - b) Write the relation(s) created in 17(a) above. (3 marks)

Print all your Queries, Tables and Relationship diagram.

Procedures (pseudocode)

```
PROCEDURE Display()
  IF (language == 1) // ENGLISH
    OUTPUT "----- OPERATIONS -----"
    OUTPUT " 1. CASH WITHDRAWAL"
    OUTPUT " 2. BALANCE INQUIRY"
    OUTPUT " 3. CANCEL"
    OUTPUT "-----";
    OUTPUT "Which operation do you want to do?"
    INPUT op
  ELSE // FRENCH
    REPEAT SAME MENU IN FRENCH
  ENDIF
  CLEAR SCREEN
  CASE OF op
    CASE 1: CALL Withdraw()
    CASE 2: CALL PrintBalance()
    CASE 3: IF (language == 1) THEN
      OUTPUT " THANK YOU FOR YOUR VISIT"
    ELSE
      OUTPUT " MERCI POUR LA VISITE "
    ENDIF
  OTHERWISE:
    IF (language == 1) THEN
      OUTPUT "Incorrect Entry\n"
    ELSE
      OUTPUT "Incorrecte."
    ENDIF
    CALL Display()
  ENDCASE
ENDPROCEDURE
```

Pseudocode for choosing language

```
OUTPUT "----- WELCOME -----"
OUTPUT "1. ENGLISH"
OUTPUT "2. FRENCH"
OUTPUT "-----"
OUTPUT "SELECT LANGUAGE:"
INPUT language

CASE OF language
  CASE 1: Call English ()
  CASE 2: Call French ()
  OTHERWISE: OUTPUT "Incorrect Entry"
ENDCASE
```

```
PROCEDURE English()
  Clear screen
  OUTPUT "Please Enter PIN:"
  INPUT pin
  DECREMENT countPin BY 1
  WHILE (PIN is not correct AND countPin > 0)
    IF (countPin > 1) THEN
      OUTPUT "Incorrect Pin. You have "&countPin
      attempt(s) left."
      OUTPUT "Please Enter PIN:"
      INPUT pin;
      DECREMENT countPin BY 1
    ENDIF
  ENDWHILE
  IF pin is correct THEN
    Clear screen
    CALL Display()
  ELSE
    OUTPUT "Your card has been blocked. Contact bank for
    more details"
  ENDIF
ENDPROCEDURE

// DO A TRANSLATION OF THE FRENCH PROCEDURE. ☺
ALL VARIABLES ARE TO BE DECLARED GLOBALLY
```

```
PROCEDURE PrintBalance()
  IF (language == 1) THEN
    OUTPUT "-----"
    OUTPUT "\t\tYOUR BALANCE IS " + GetBalance() + " FCFA"
    OUTPUT "-----"
    OUTPUT "Do you want to do another
    Operation? (Yes/No) "
    INPUT ans
    IF (ans == "Yes") THEN
      Display();
    ELSE
      OUTPUT "Thank you for your Visit."
    ENDIF
  ELSE
    // TRANSLATE TO FRENCH THE ABOVE CODE AND INSERT
    BELOW
  ENDIF
ENDPROCEDURE
```

```
FUNCTION GetBalance()
  // withdraw the money
  FOR i = 0 to 5
    IF (pin == Account[i][1]) THEN
      return Account[i][2]
    ENDIF
  ENDFOR
ENDFUNCTION
```