

Paper 1

1	B	11	B	21	B	31	C	41	A
2	A	12	C	22	A	32	D	42	B
3	C	13	B	23	A	33	A	43	A
4	A	14	C	24	C	34	C	44	B
5	D	15	C	25	C	35	B	45	B
6	B	16	B	26	B	36	D	46	D
7	C	17	B	27	C	37	B	47	B
8	C	18	C	28	B	38	B	48	D
9	C	19	B	29	A	39	A	49	C
10	C	20	C	30	D	40	C	50	A

Paper 2

Question 1

- i) a) $\bar{x}\bar{y} + x.y = \overline{\bar{x}\bar{y} + x.y}$ } **1 mark**
 $= \overline{\bar{x}\bar{y}} . \overline{x.y}$
 $= (x + y).(\bar{x} + \bar{y})$
 $= x(\bar{x} + \bar{y}) + y(\bar{x} + \bar{y})$ } **1 mark**
 $= x\bar{y} + \bar{x}y$
- b) $f = \bar{x}. \bar{y}. z + \bar{x}. y. \bar{z} + \bar{x}. \bar{y}. \bar{z} + x. y. \bar{z}$
 $= (\bar{x}. \bar{y}. z + \bar{x}. \bar{y}. \bar{z}) + (\bar{x}. y. \bar{z} + x. y. \bar{z})$ **1 mark**
 $= \bar{x}. \bar{y} + y. \bar{z}$ **1 mark**
- c) - Contain fewer gates than the original circuit.
- Reduction of cost and the overall complexity.
- Reduces number of wires. } **Any two (2 mark)**
- ii) a)

Basic	CISC	RISC
Instruction Set	Complex instructions	Simple instructions
Program Code Size	Smaller	Lengthier
Processor Size	Increased hardwired circuitry leads to increased processor size	Reduced hardwired circuitry leads to reduced processor size
Memory Usage	Less memory intensive	More memory intensive
Power Consumption	More power	Less power
Heating	More heat	Less heat
	Fixed length instructions	Variable length instructions

SRAM	DRAM
Transistors are used to store information	Capacitors are used to store data
Faster access speeds	slow access speeds.
It does not have a refreshing unit.	It has a refreshing unit.
These are expensive.	These are cheaper.
Low-density devices.	High-density devices.
Bits are stored in voltage form.	Bits are stored in the form of electric energy.
Used in cache memories.	Used in main memories.
Consumes less power and generates less heat.	Uses more power and generates more heat.

b)

Stage	Time Interval								
	1	2	3	4	5	6	7	8	9
Fetch Instruction	A	B	C						
Decode Instruction		A	B	C					
Execute Instruction			A	B	C				
Access operand in memory				A	B	C			
Write result to register					A	B	C		

- With pipelining: no of cycles = 7
- Without pipelining: no of cycles = $3 * 5 = 15$
- No of cycles saved = $15 - 7 = 8$

iii)

a)

Characteristics	Unit of measurement
Storage capacity: Storage capacity refers to the size of the memory, that is, the amount of data that can be stored in the memory.	bits or bytes
Cost: The cost of memory is valued by estimating the cost per bit of storage, that is, the cost of a storage unit for a given storage capacity.	Cost per bit of storage
Transfer Rate: the transfer rate is the number of characters or words that a device can transfer per second.	bits per second or bytes per second

Other characteristics are: Mutability, accessibility, addressability and volatility.

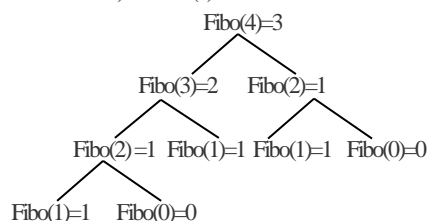
b)

Sequential Access	Random Access
Any location in storage can be accessed at any moment in approximately the same amount of time. Examples are primary and secondary storage devices (RAM, hard disk drives, solid state drives, removable "USB" drives, CDs, and DVDs)	The accessing of pieces of information will be in a serial order, one after the other; therefore, the time to access a particular piece of information depends upon which piece of information was last accessed. Examples are offline storage devices (floppy disks, compact disks and USB sticks)

QUESTION 2

i)

- a) 8th term = 21.
b) Fib(4) = 3



d)

```

FUNCTION Main
  counter = 0
  Output "Enter the value of n:"
  Input n
  Output "Fibonacci Series:"
  FOR i = 1 to n
    Output fibo(counter)
    counter = counter + 1
  ENDFOR

```

ii)

a)

PIN	C	x	Message
51020	0	51020	
	1	5102	
	2	510.2	
	3	51.02	
	4	5.102	
	5	0.5102	"PIN OK"
5120	0	5120	
	1	512	
	2	51.2	
	3	5.12	

c)

```

Function fibo (Integer n)
  Declare Integer Array f[n+2]
  Declare Integer i, f0, f1, fn
  Assign f0 = 0
  Assign f1 = 1
  For i = 2 to n
    Assign fn = f0 + f1
    Assign f0 = f1
    Assign f1 = fn
  End
  Return Integer f1
ENDFUNCTION

FUNCTION fibo (Integer n)
  IF (n=0 OR n=1) THEN
    Return n
  ELSE
    Return fibo(n-1) + fibo(n-2)
  ENDIF
ENDFUNCTION

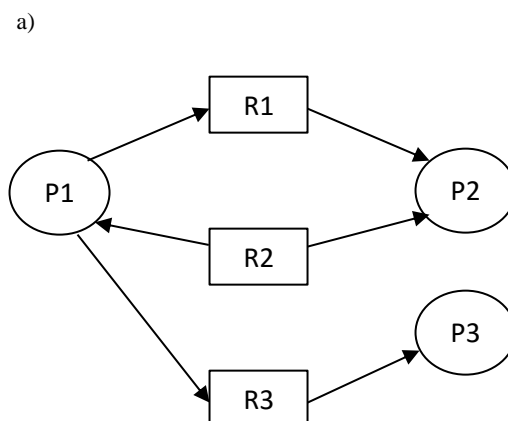
```

- | | | | |
|--|---|-------|------------------------|
| | 4 | 0.512 | “error in PIN entered” |
|--|---|-------|------------------------|
- **51020:**
value of c: 5 (1 mark)
message: PIN OK (1 mark)
 - **5120:**
value of c: 4 (1 mark)
message: error in PIN entered (1 mark)
 - b) Length check
 - iii) 1 mark for each error identified + suggested correction
line 6: this should read **h = x**
line 8: **PRINT h** should come after the end of the repeat loop
line 9: this should read **UNTIL c = 20 or UNTIL c >= 20 or UNTIL c > 19**

QUESTION 3

- i)
 - a) - Distributed OS: multiple computers linked by communications and remote access is enabled within the devices connected in that network. It is an extension of Network OS.
 - Real time OS: Process control is the ability to monitor and adjust a process to give a desired output. Process is done within a short period of time.
 - Batch OS: Group of tasks processed at once.
 - b) Efficient resources utilization
 - c) An interrupt is a signal emitted by hardware or software when a process or an event needs immediate attention.

ii)



- b) No deadlock exists.
 - the graph does not contain a cycle.
 - P1 and P2 are holding the same resource R2.
- c) Abort a process or preempt some resources when deadlocks are detected.

iii)

a)

Round robin

0	4	8	12	16	20	24	25
P ₁	P ₂	P ₃	P ₄	P ₁	P ₃	P ₄	P ₃
4	8	12	16	20	24	25	26

Shortest job time

0	8	12	17
P ₁	P ₂	P ₄	P ₃
8	12	17	26

b) Round robin

Average waiting time = ((20-8) + (8-4-1) + (26-9-2) + (25 -5-3))/4 = (12+3+15+17)/4 = 47/4 = 11.75

Shortest job time

Average waiting time = (0-0) + (8-1) + (17-2) + (12-3))/4 = (0+7+15+9)/4 = 26/4 = 7.75

QUESTION 4

i)

- a) A flat file database is a type of database that stores data in a single table.
- b) Referential integrity refers to the accuracy and consistency of data within a relationship. That is, whenever a foreign key value is used it must reference a valid, existing primary key in the parent table.

Example:

User			Review	
Userid	Username	Password	Userid	Restid
1	Jeskovic	grandista	1	2
2	Manasta	Resqui12	1	3

We can see that the foreign key value of Userid 1 is found in the parent table User.

- c) Any two
- restricting access to certain users,
 - controlling what each user can do and
 - running anti-virus software.
- ii) a) Award up to [2 max, 1 mark per point].
- Use a text field for the Supp_Phone with a length of 15 characters (*length check*) to ensure that an entered phone number is not shorter or longer than 15 characters. Entering fewer or more characters makes a phone number invalid.
 - Use a validation rule (format check) to ensure the first six characters can only be “(+237)”
 - Use an input mask such as LLLLLL000000000 (L represents any letter and 0 represents any digit) to ensure the last 9 entries are only numbers. For example (+237) 6542 98001 will be accepted as valid.

b) Award up to [2 max].

Prod_ID	Prod_Name	Prod_Price	Supp_Name	Supp_Contact	Supp_Phone
P211	Apples	2625	Fruit and Veggie	Ines Bond	(+237) 6542 98001
P121	Onions	1575	Fruit and Veggie	Ines Bond	(+237) 6542 98001

c) Award up to [1 max].

- When inserting a tuple supplied by an existing supplier, it may lead to *inconsistency* if the data entered is not consistent with the data about this supplier in all other tuples;

For example, inserting:

P189	Potatoes	2700	Veggy Co.	Mia Abiss	(+237) 6234 76547
------	----------	------	-----------	-----------	-------------------

Award up to [1 max].

- Deleting a tuple could remove data which is not intended to be lost;

For example; deleting the following tuple.

P265	Tomatoes	4000	New Fruits	John Smith	(+237) 677 789521
------	----------	------	------------	------------	-------------------

Which is the only tuple in the relation which holds data about New Fruits suppliers (all data about this supplier will be lost); [1]

Award up to [1 max].

- Modifying/changing the phone number for Fruit and Veggie. would require that each and every tuple containing item supplied by Fruit and Veggie should be modified (unnecessary work);

Prod_ID	Prod_Name	Prod_Price	Supp_Name	Supp_Contact	Supp_Phone
P211	Apples	2625	Fruit and Veggie	Ines Bond	(+237) 6542 98001
P121	Onions	1575	Fruit and Veggie	Ines Bond	(+237) 6542 98001

d) Award up to [4 max]

3NF
SupplierContact(Supp_Phone, Supp_Contact, Supp_ID*)
Supplier(Supp_ID, Supp_Name)
Product(Prod_ID, Prod_Name, Prod_Price, Supp_ID*)

QUESTION 5

i) a) Any two from:

- install software and hardware
- transfer of files to new system
- training of the staff/workers
- production of documentation
- remove the old software

b.i)

Parallel

- both the old system and new system are run together
- two sets of workers needed to run both systems together

Pilot

- new system is run in one warehouse of the company
- once the new system is shown to be OK, it is rolled out to rest of company

- if new system fails there is still the old system as backup
- possible to train work force on new system while old system still running
- compare old and new systems to ensure new system is working correctly.

Pilot

- if old system fails, only one warehouse affected
- training can be developed in “pilot” warehouse for rest of company

Not training as both examples

- c) 1 mark for description, 1 mark for suitable example
- Corrective
- solve any bugs/problems in the software not discovered at testing
- Example: user reports a problem

Must give answers from both methods

b.ii)

Parallel

Adaptive

- alter the solution to take into account changes brought about by external/internal Influences

Example: new laws, new legislation, etc.

ii)

- a) One mark for each correct check (max two)
- Length (check)

QUESTION 6

i)

- a) $10010111.0100 = 151.25$

Binary	10010111	0.0100
Denary	151	0.25

- b) 10010111.0100 to 2's compliment = 105.6875

Binary number	10010111.0100
1's compliment	01101000.1011
2's compliment	$01101000.1011 + 1 = 01101000.1100$
Denary	-104.75

c)

Mantissa: 1.0010111

-		$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$	$\frac{1}{128}$
1	.	0	0	1	0	1	1	1

exponent: 0100

8	4	2	1
0	1	0	0

Since the mantissa has a negative value, first convert the value using two's complement.

So, the mantissa will be $01101000 + 1 = 01101001$.

Therefore, mantissa is -0.1101001

Exponent is 4, so move the binary point four places to the right (to match the positive exponent value). This gives -1101.001

-8	-4	-2	-1	.	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$
----	----	----	----	---	---------------	---------------	---------------

- b) Any two correct (1 mark each)

Von Neumann architecture	Harvard architecture
It is ancient computer architecture based on stored program computer concept.	It is modern computer architecture based on Harvard Mark I relay based model.
Same physical memory address is used for instructions and data.	Separate physical memory address is used for instructions and data.
There is common bus for data and instruction transfer.	Separate buses are used for transferring data and instruction.
Two clock cycles are required to execute single instruction.	An instruction is executed in a single cycle.
It is cheaper in cost.	It is costly than Von Neumann Architecture.
CPU cannot access instructions and read/write at the same time.	CPU can access instructions and read/write at the same time.
It is used in personal computers and small computers.	It is used in micro controllers and signal processing.

- c) Any two. 1 mark each

- Two separate areas of memory: one for instructions & one for data/instructions and data can be accessed concurrently.
- Different (sets of) buses: one for instructions & one for data/instructions and data can be accessed concurrently.

- Type Check
- Format Check

- b) One mark for each suitable piece of test data and one mark for each relevant reason (max four)

- LL9999LL999: Too long
- 5678987: All numeric
- CB12EU: No space is present

1	1	0	1		0	0	1
---	---	---	---	--	---	---	---

Therefore, -1101.001 in denary is $-8-4-1-0.125 = -13.125$

ii)

- a) The components are the following: any three 1 mark each

- **Control Unit** – Handles all processor control signals. Directs all input and output flow, fetches code for instructions, and controls how data moves around the system.
- **Arithmetic and Logic Unit (ALU)** – Handles all the calculations the CPU may need. That is arithmetic operations (addition, subtraction, etc.) and logical operations (AND, OR, NOT, etc.)
- **Input Devices** – Program or data is read into main memory from the *input device* or secondary storage under the control of CPU input instruction.
- **Output devices** - used to output information from a computer.
- **Registers** – Registers refer to high-speed storage areas in the CPU. The data processed by the CPU are fetched from the registers.
- **Memory unit** - Sometimes referred to as primary or main memory. Unlike a hard drive (secondary memory), this memory is fast and also directly accessible by the CPU.

- Virtual cores/Hyper-threading™: Treating a physical core as two virtual cores.
 - Multiple Cores: Each core acts as a separate processing unit.
 - Onboard Graphics: Built in circuitry for graphics processing.
- iii) 1 character = 1 byte = 8 bits
- a) So:
Name = $20 * 8 \text{ bits} = 160 \text{ bits}$

QUESTION 7

- i)
- a) Any TWO
- All computers are of equal status
 - Each computer provides access to resources and data
 - Computers can communicate and share resources
 - Each computer is responsible for its own security
- b) Any Two
- Reduced security // no central management of security
 - only as secure as the weakest computer on the network
 - each computer is at risk from viruses from other computers
 - No central management of backup
 - if the data from one computer is not backed up it is lost to all of them
 - No central management of files/software
 - consistency may be difficult to maintain
 - each computer may have different software from the others
 - Individual computers may respond slower
 - because they are being accessed by other computers
 - In order to share files etc, all the computers involved need to be switched on so the files etc. may not be always available.
- c) ✓
- receives packets from devices
 - stores the IP and/or MAC address of all devices attached to it

QUESTION 8.

- i)
- a) An information system is the collection of software, hardware, networking infrastructure, human resources and databases that provide the storage, processing and communication/distribution of information (1 mark) to support planning, control and decision making in an organization (1 mark). (Award up to max 2)
- b) Transaction Processing System: (Award up to max 2)
- A TPS is the basic information system that serves the operational level of the organization.

Date of birth = $10 * 8 \text{ bits} = 80 \text{ bits}$
 Sex = $1 * 8 \text{ bits} = 8 \text{ bits}$
 Height = 16 bits
 Therefore, the variable T = $16 + 8 + 80 + 160 = 264 \text{ bits} = 264 / 8 = 33 \text{ bytes}$

b) 1 kilobyte = 1024 bytes
 Therefore, 33 bytes = $33 / 1024 \text{ kilobytes} = 0.032 \text{ kilobytes}$

✓

Wired

Faster connection // higher bandwidth

- needed as she is downloading/streaming large files
- ... less time waiting / less latency / fewer delays
- More reliable / stable connection
- ... is less susceptible to issues with distance/walls/interference
- More secure Wireless
- Freedom of movement

d)

(i) 1 mark for identifying that she is using both.*1 mark per bullet point for justification*

- using internet because sending data on the infrastructure
- using WWW because accessing a website (that is stored on a web server operated by the webmail) that is part of the WWW

(ii) The complexity of an algorithm defines the performance of the algorithm in terms of the input size. We consider the complexities of every algorithm and compare them while choosing the most efficient algorithm to solve our problem.

(iii) Divide and conquer algorithm operates in three stages:

Divide: Divide the problem recursively into smaller subproblems.

Solve: Subproblems are solved independently.

Combine: Combine subproblem solutions in order to deduce the answer to the original large problem.

- It captures and processes data generated during an organization's day-to-day transactions and maintains records about the transactions.
- A transaction is any activity of an organization including a business activity such as a *deposit, payment, order or reservation*.
- TPS are vital for any organization or business as they gather all the input necessary for other types of systems.

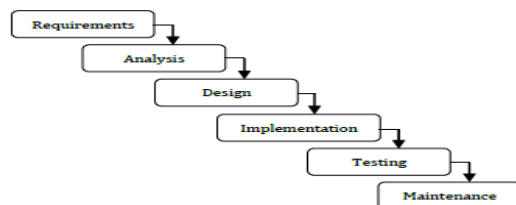
ii)

- a) In evolutionary prototyping, a functional prototype with minimal functionality is built in the beginning

and is refined over time, as requirements are better understood while in incremental prototyping, functional prototypes of the various subsystems are built and then integrated to form a complete system, that is, the product is built as separate prototypes which are later merged into a final product.

b) (Award max 2)

- The waterfall model is a systematic, sequential approach to software development that begins with the system requirements and progresses through analysis, design, coding, testing, and support
- In the waterfall model, each phase must be completed in its entirety before the next phase can begin.
- At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project.
- In waterfall model phases do not overlap.



Limitations (Award max 2)

- Adjusting scope during the life cycle can kill a project
 - No working software is produced until late during the life cycle.
 - High amounts of risk and uncertainty.
 - Poor model for complex and object-oriented projects.
- iii)
- a) (Award Max 2) **1 mark each per technique and explanation (2 × 1)**
- **Interview:** An interview is a meeting during which a person is asked questions. Interviews allow system analyst to collect a greater depth of data and understanding from people than is possible by just using a questionnaire.

- The system analyst can interview a manager of a cafeteria or the ‘Healthy Eating’ expert or cafeteria staff. The healthy eating expert can give the system analyst information on what constitute a balanced diet
 - The system analyst can also interview (a sample of) pupils who are currently using the old system to understand how it functions;
 - **Questionnaires:** A questionnaire is a set of questions in paper format or softcopy used in a survey to collect data from people.
 - A questionnaire can be developed for pupils and another for staff asking them about their views concerning the current system been used.
 - **Observation:** In observation, the system analyst observes what is happening during a process or event and produces a data file as a result
 - The system analyst can observe the current system to understand how cafeterias are being used and how food items can be labelled;
- b) (Award Max 3) **1 mark each** for 3 sensible points
3 × 1
- Review design or coding against spec.
 - Produce test strategy
 - Test and correct / debug
 - Decide on rollout / changeover strategy
 - Produce user and/or technical documentation (can be 2 points)
 - Load data into the new system
 - Train users.
- c) (Award Max 2) **0.5 mark each** for 4 sensible points
4 × 0.5
- Log of error frequency;
 - Log of breakdown / maintenance incidents;
 - Feedback from user / measure of user satisfaction / ease of use;
 - Analysis of costs compared with old system;
 - Comparison with original objectives for effectiveness,
 - Observation of specific task improvements;

Paper 3

Task 1

1. Following are the input and output devices used
 - Input devices: Keypad, touchpad (touch screen), card reader (1 mark)
 - Output devices: Display Screen, Receipt Printer, speaker, cash depositor. (1 mark)
2. Verification check
 - Double Entry (1 mark)
3. Validation checks
 - Length check: A length check will verify that the number of digits/characters meets set expectations. E.g. PIN must be four digits. (2 marks)

Task 2

4. This is the code of the menu in C programming language:

```
//INSERT LANGUAGE MENU HERE
printf("----- WELCOME -----\\n");
printf("1. ENGLISH\\n");
printf("2. FRENCH\\n");
printf("-----\\n");
printf("SELECT LANGUAGE:");
scanf("%d", &language);

switch(language)
{
    case 1: English();
            break;
    case 2: French();
            break;
    default: printf("Incorrect Entry\\n");
}
}
```

5. Below are the procedures

```
void English()
{
    system("cls");
    printf("Please Enter PIN:");
    scanf("%d",&pin);
    countPin--;
    while(!isincorrectpin() && countPin > 0)
    {
        if (countPin > 1)
        {
            printf("Incorrect Pin. You have %d attempt(s) left.",countPin);
            printf("Please Enter PIN:");
            scanf("%d",&pin);
            countPin--;
        }
    }
    if (isincorrectpin())
    {
        system("cls");
        Display();
    }
    else
    {
        printf("Your card has been blocked. Contact bank for more details");
    }
}
}
```

```
void French()
{
    system("cls");
    printf("SVP Entrez votre PIN:");
    scanf("%d",&pin);
    countPin--;
    while(!isincorrectpin() && countPin > 0)
    {
        if (countPin > 1)
        {
            printf( "Pin Incorrecte. Vous avez %d essai(s).\\n",countPin);
            printf( "SVP Entrez votre PIN:");
            scanf("%d",&pin);
            countPin--;
        }
    }
    if (isincorrectpin())
    {
        system("cls");
        Display();
    }
    else
    {
        printf("Votre carte a ete bloquee. Pour plus d'information, contactez votre banque\\n");
    }
}
}
```


6.

```

int Display()
{
    if (language == 1)//ENGLISH
    {
        printf("----- OPERATIONS -----\\n");
        printf(" 1. CASH WITHDRAWAL\\n");
        printf(" 2. BALANCE INQUIRY\\n");
        printf(" 3. CANCEL\\n");
        printf("-----\\n");
        printf("Which operation do you want to do?\\n");
    }
    else // FRENCH
    //REPEAT SAME MENU IN FRENCH
    {
        printf("----- OPERATIONS -----\\n");
        printf(" 1. RETRAIT\\n");
        printf(" 2. SOLDE\\n");
        printf(" 3. ANNULER\\n");
        printf("-----\\n");
        printf("Quelle operation voulez vous effectue?\\n");
    }
    scanf("%d",&op);
    system("cls");
    switch(op)
    {
        case 1: Withdraw();
            break;
        case 2: PrintBalance();
            break;
        case 3: if (language == 1)
                printf(" THANK YOU FOR YOUR VISIT\\n");
            else
                printf(" MERCI POUR LA VISITE\\n");
            break;
        default:if (language == 1)
                printf("Incorrect Entry\\n");
            else
                printf(" Incorrecte.\\n");
            Display();
    }
}

```

7.

```

void Withdraw()
{
    time_t t;
    time(&t);
    TransNo = (rand() % 999999) + 100000;
    No = (rand() % 999) + 100;

    if (language == 1)
    {
        //INSERT ENGLISH VERSION OF THE MENU AS SHOWN IN THE DIAGRAM FOR WITHDRWAL
        printf("-----\\n");
        printf(" 1. CFA 5000 F   2. CFA 50000 F\\n");
        printf(" 3. CFA 10000 F  4. CFA 100000 F\\n");
        printf(" 5. CFA 20000 F  6. CFA 150000 F\\n");
        printf(" 7. CFA 40000 F  8. OTHER AMOUNT\\n");
        printf("-----\\n");
        printf("    CHOOSE AMOUNT:");
    }
}

```



```

else
{
    //INSERT FRENCH VERSION OF WITHDRAWAL MENU
    printf("-----\n");
    printf(" 1. CFA 5000 F   2. CFA 50000 F\n");
    printf(" 3. CFA 10000 F  4. CFA 100000 F\n");
    printf(" 5. CFA 20000 F  6. CFA 150000 F\n");
    printf(" 7. CFA 40000 F  8. AUTRE MONTANT\n");
    printf("-----\n");
    printf("  ENTREZ VOTRE MONTANT:");

}
scanf("%d",&op);
//system("cls");

switch(op)
{
    //INSERT VARIOUS CASES FOR AMOUNT ENTERED OR CHOSEN
    case 1: amt = 5000;
        break;
    case 2: amt = 50000;
        break;
    case 3: amt = 10000;
        break;
    case 4: amt = 100000;
        break;
    case 5: amt = 20000;
        break;
    case 6: amt = 150000;
        break;
    case 7: amt = 40000;
        break;
    case 8: if (language == 1)
        printf("Amount:");
        else
        printf("Montant:");

        scanf("%d",&amt);
        break;
    default:if (language == 1)
        printf("Incorrect Entry!!!\n");
        else
        printf("Incorrecte!!!");
        //system("cls");
        Withdraw();
}
//check the amount entered if superior to account balance
if (!isincorrectamt(amt))
{
    if (language == 1)
        printf("Sorry, Insufficient Funds.\n\n");
    else
        printf("Desole, Solde Insuffisant.\n\n");

    Display();
}
else
{
    //withdraw the money
    for (i=0;i<5;i++)
    {
        //GET THE POSITION OF THE CUSTOMER'S NAME AND ACCOUNT NO USING A
        CONDITIONAL STATEMENT
        //if pin equal to any pin in the Account array
        if (pin == Account[i][1])
        {
            //ASSIGN THE NEW BALANCE WHICH IS AVAILABLE BALANCE – AMT – FEE

```

}

8.

```

int PrintBalance()
{
    if (language == 1)
    {
        printf("-----\n");
        printf("\t\tYOUR BALANCE IS %d FCFA\n",GetBalance());
        printf("-----\n");
        printf("Do you want to do another Operation?(Yes/No)\n");
        scanf("%s",&ans);
        if (strcmp(ans,"Yes")==0)
            Display();
        else
            printf("Thank you for your Visit.\n");
    }
    else
    {
        //TRANSLATE TO FRENCH THE ABOVE CODE AND
        INSERT BELOW
        printf("-----\n");
        printf("\t\tVOTRE SOLDE EST %d FCFA\n",GetBalance());
        printf("-----\n");
        printf("Voulez vous faire une autre transaction?(Oui/Non)\n");
        scanf("%s",&ans);
        if (strcmp(ans,"Oui")==0)
            Display();
        else
            printf("Merci pour votre visite.\n");
    }
}

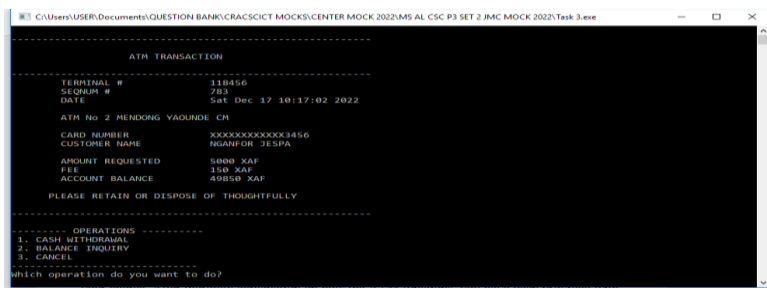
```

```

int GetBalance()
{
    //withdraw the money
    for(i=0;i<=5;i++)
    {
        if (pin == Account[i][1])
            return Account[i][2];
    }
}

```

9.



10. CREATE DATABASE CLUB;

```

11. CREATE TABLE Course(
    CourseId Varchar(6) PRIMARY KEY,
    Title Varchar(30) not Null,
    Duration int not Null,
    Fee money,
    CourseType Varchar(7)
);
1 mark for SQL statement and 1 mark for any two tables created.

```

12. See printouts of relationship.

13. Check from printout.

```

14. INSERT INTO Course VALUES ("GCEC02", "GCE-Computer-FT", 12, 5000, "Monthly");

```

```

15. UPDATE Course SET Title = "GCE-Computing-ST"
WHERE Title= " GCE-Computer-FT";

```

```

16. SELECT Student.Forename, Student.Surname
FROM Student, Course, Teacher, CourseEnrolled
WHERE Student.StudentId=CourseEnrolled.StudentId
AND
Course.CourseId=CourseEnrolled.CourseId AND
Teacher.CourseId=CourseEnrolled.CourseId
AND courseType="Weekly" AND
Teacher.Forename="William" AND
Teacher.Surname="Smith" Order By Student.Forename

```

17.

a) Create a new table Parents with primary key ParentId which will be added into students table as foreign key as there will be one parent record will have many child records in the student table.

b)

- Student (StudentId, ForeName, SurName, DOB, Address, City, ParentId) **1 mark**
- Parent (ParentId, ForeName, SurName, DOB, Address, City) **2 marks**

