PF LAB 02 TASK

1. Design a flowchart, Pseudocode, Algorithm for processing a customer order at a restaurant, including handling special requests (Like add on).

- Pseudo code:-

1. Start
2. Display “welcome to KFC. How may I help you?”
3. Display menu to customer
4. Read order
5. Input order
6. Read add on
7. Input add on
8. Calculate bill
9. Display bill
10. Take cash
11. Display wait time “sir, your order will be ready in 10 mins”
12. end

Algorithm:-

1. greet the customer
2. give the menu to customer
3. ask the customer their order
4. enter the customers order 1
5. ask the customer for add ons
6. enter the customers add on 1
7. set the bill to order 1 + add on 1
8. display the bill to the customer
9. get cash from the customer
10. display the wait time

2. Design a flowchart, Pseudocode, Algorithm for handling a customer's deposit transaction at a bank, including checks for account validity and deposit amount conditions.

-pseudocode:-

- start
- display “please put your card in the card slot”
- read credit card
- display “enter pin”
- read pin
- IF pin is correct,
    - Display multiple transaction options
    - Read “cash withdrawal”
    - Display “please enter withdrawal amount”
    - Read amount
    - IF amount<=bank balance
        - Display “please wait, processing your cash”
        - Display “please take your card”
        - Display “please take your cash”
        - Display “please take you receipt”
        - end
    - ELSE display “insufficient funds. Please enter a different amount”
    - REPEAT steps 10-17
    - Until amount<=bank balance
    - Repeat steps 12-15
    - end
- ELSE display “incorrect pin. Please enter the correct pin”
- Repeat
- steps 5-22
- Until entered pin==correct pin
- Repeat
- steps 7-19
- end

Algorithm:-

- ask the user to input their credit card
- read the users credit card
- ask the users pin
- read users inputted pin
- IF pin is correct,
    - Show user multiple transaction options
    - Read users selection of cash withdrawal option
    - Ask user the amount to withdraw
    - IF the amount is less than or equal to bank balance
        - Give the user their card
        - Give the user their cash
        - Give the user their receipt
    - ELSE repeat steps 8-12 until the amount is within the bank balance
- ELSE ask the user to input the correct pin
- Repeat steps 4-14 until the inputted pin is same as correct pin
- Repeat steps 6-13

3. Design a flowchart, Pseudocode, Algorithm to determine which of three provided numbers is the greatest.

-pseudocode:-

- start
- read numbers x, y, z
- display “which number is the greatest”
- set x=greatest number
- IF y>x
    - Set y= greatest number
- ENDIF
- IF z>y

    - Set z= greatest number
- ENDIF
- Print greatest number
- end

Algorithm:-

- ask the user to input 3 numbers x, y, z
- read the three numbers
- set number x as the greatest
- IF y>x
    - Set number y as the greatest
- END IF
- IF z>y
    - Set number z as the greatest
- END IF
- Display to the user the greatest number

4. Implement an algorithm where the user enters a number, and an appropriate month is displayed.

-algorithm:-

- Ask the user to input a number
- Read the number
- IF number==1
    - Display the month of January
    - ELSEIF number==2
    - Display the month of February
    - ELSEIF number==3
    - Display the month of March
    - ELSEIF number==4
    - Display the month of April
    - ELSEIF number==5
    - Display the month of May
    - ELSEIF number==6
    - Display the month of June

  - ELSEIF number==7
  - Display the month of July
  - ELSEIF number==8
  - Display the month of August
  - ELSEIF number==9
  - Display the month of September
  - ELSEIF number==10
  - Display the month of October
  - ELSEIF number==11
  - Display the month of November
  - ELSEIF number==12
  - Display the month of December
- ENDIF
- Display to the user the appropriate month.

5. Create pseudocode a small calculator which only does '+' or '-'Operations. (Hint: Take three variable inputs with one being used for the operator)

- Start
- Display "input N1"
- Read N1
- Display "input N2"
- Read N2
- Display "input operator '+' or '-' "
- Read operator
- IF operator==+
  - Result=N1+N2
  - Print result
  - ELSEIF operator==-
  - Result=N1-N2
  - Print result
- ELSE display "please input a valid operator either '+' or '-'"
- ENDIF
- End

7. Implement an algorithm for making a simple calculator with all the operators (+,-,*,/,%)

- Ask the use to input first number
- Ask the user to input an operator from (+,-,*,/,%)
- Ask the user to input a second number
- IF the operator==+
    - Set result=N1+N2
    - ELSEIF operator==-
    - Set result=N1-N2
    - ELSEIF operator==*
    - Set result=N1*N2
    - ELSEIF operator==/
    - Set result=N1/N2
    - ELSEIF operator==%
    - Set result=N1%N2
- ELSE tell the user to input a proper operator from (+,-,*,/,%)

9. Why we use .gitignore?

- .gitignore is used to ensure that certain files remain untracked.

10. Difference between Algorithm and Pseudocode?

- **algorithm** consists of sets of instructions for the user to perform

Pseudocode is a way of expressing an algorithm in the form of a programming language but more simpler.