**CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY**

**Chandubhai S Patel Institute of Technology**

Department of Computer Science & Engineering

|  |  |
| --- | --- |
| **No. of program** | **Program** |
| |  | | --- | | **1.1** | | |  | | --- | | Write a C program that will output this passage by Michael Singer. Make sure your output looks exactly as shown here (including spacing, line breaks, punctuation, and the title and author). Use Required Escape Sequence and ASCII Value. | |
| |  | | --- | | **Flowchart** | |  |
| |  | | --- | | **Algorithm** | | |  | | --- | | Step 1: Start.  Step 2: Declare Smile, Diamond, Heart.  Step 3: Print Ascii Values of Smile, Diamond and Heart 37 Times.  Step 4: Print Ascii Value, Text, Ascii Value 4 Times.  Step 5: Print Ascii Values of Smile, Diamond and Heart 37 Times.  Step 6: Stop. | |
| **Code** |  |
| **Output** |  |
| **Question** | |  |  |  | | --- | --- | --- | | Sr. No. | Symbol | Ascii Value | | 1 | 😊😊😊 | 002 | | 2 | ••• | 007 | | 3 | >>> | 062 | | 4 | <<< | 060 | | 5 | === | 061 | |

|  |  |
| --- | --- |
| **1.2** | Write your bio-data using Escape Sequences. And you have to take  your Basic  Information as user input. It should contain the following content. It should contain the following content. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare a=61, b=35, c=64, d=58.  Step 3: print value of ‘a’ and ‘b’.  Step 4: print Basic Information.  Step 5: Print Education and Other Information.  Step 6: print value of ‘a’ and ‘b’ as “step 3” time.  Step 7: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. What is the purpose of using escape sequences? Answer in one or two statements. Mention any 5 escape sequences used regularly along with their purpose.  |  |  |  | | --- | --- | --- | | Sr.No. | Escape Sequences | puspose | | 1 | \n | Enter in new line | | 2 | \t | Words that come after ‘\t’ will be pushed in the same line leaving some spaces | | 3 | \v | This is used to print the vertical tab | | 4 | \r | This is the escape sequence to position the cursor at the beginning of the line | | 5 | \a | This is the escape sequence to generate a bell sound to denote the execution of the program | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2.1** | In a town, the percentage of men is 52. The percentage of total literacy is 48. If total percentage of literate men is 35 of the total population, write a program to find the total number of illiterate men and women if the population of the town is 80,000.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Fill below mentioned table as per your output.   |  |  |  | | --- | --- | --- | | Sr.no. | Get Outcome | value | | 1 | Total population |  | | 2 | Number of Literate (Men + Women) |  | | 3 | Number of Men |  | | 4 | Number of Literate Men |  | | 5 | Number of illiterate Men |  | | 6 | Number of Women |  | | 7 | Number of Literate Women |  | | 8 | Number of illiterate Women |  | |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare Total Population, Total Men, Total Literacy, Literate Men, Total Women, Literate Women, Illiterate Men, Illiterate Women.  Step 3: TotalPopulation80000  Step 4: Total Men = Population\*(0.52)  Step 5: Literate Men = Total Population\*(0.35)  Step 6: Total Literacy = Total Population\*(0.48)  Step 7: Illiterate Men = Total Men-Literate Men  Step 8: Total Women = Total Population - Total Men  Step 9: Literate Women = Total Literacy- Literate Men.  Step 10: Illiterate Women = Total Women- Literate Women  Step 11: Print Total Population, Total Men, Literate Men, Total Literacy, Illiterate Men, Total Women, Literate Women, Illiterate Women.  Step 12: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Has this scenario helped you learn about integer and float datatype? If yes, then mention the requirements of using integer and float data types.   **Answer**: Yes,Integer and float use for store “number” data type. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2.2** | A Bigbazaar cashier has currency notes of denominations 10,50 and 100. If the amount to be withdrawn is input through the keyboard in hundreds, find the total number of currency notes of each denomination the cashier will have to give to the withdrawer.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output  Fill up the required number of currency notes of denomination 10, 50 and 100 in below given table as per the output received.   |  |  |  | | --- | --- | --- | | Sr. No. | Note Requirements | Counts | | 1 | Requirement of 100 Rs. note |  | | 2 | Requirement of 50 Rs. note |  | | 3 | Requirement of 10 Rs. note |  | |
| **Flowchart** |  |
| **Algorithm** | Step1: Start  Step2: Declare Amount, HundredNotes,FiftyNotes,TenNotes  Step3: Collect Value Of Amount From The User  Step4: HundredNotesAmount/100  Step5: Print HundredNotes  Step6: AmountAmount%100; FiftyNotes=Amount/50;  Step7:Print FiftyNotes  Step8:AmountAmount%50; TenNotes=Amount/10;  Step9:Print TenNotes  Step10:Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you learned about how scanf function can be used to collect the user input? Give the correct answer for the following table:  |  |  |  |  | | --- | --- | --- | --- | | Sr. No. | Data types | Format specifier | Example data | | 1 | Integer | %d | ‘2’, ‘44’, ‘87’ | | 2 | Float | %f | ‘4.3’,5.866’, | | 3 | char | %c | ‘a’, ‘b’, | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2.3** | Write a program to calculate Net Salary. User has to input Basic Salary and Output  should be:  Enter Basic Salary: 5000 (e.g. 5000)  Allowances:  DA = 70% of Basic Salary  HRA = 7% of Basic Salary  MA = 2% of Basic Salary  TA = 4% of Basic Salary  Deduction:  PF = 12% of Basic Salary  IT = any value (e.g. 500)  ----------------------------------------------------------------------------------------------  Net Salary = Basic Salary + Allowances – Deduction  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Fill up the data mentioned in below given table as per the output received.     |  |  |  | | --- | --- | --- | | Sr. No. | Input/Outputs  Amount | Amount | | 1 | Enter your Basic Salary |  | | 2 | DA of Basic Salary |  | | 3 | HRA of Basic Salary |  | | 4 | MA of Basic Salary |  | | 5 | TA of Basic Salary |  | | 6 | PF of Basic Salary |  | | 7 | Gross Salary |  | | 8 | Net Salary |  | |
| **Flowchart** |  |
| **Algorithm** | Step 1: start  Step 2: Salary, Net Salary, DA, HRA, MA, TA, PF, IT, Allowances, Deduction  Step 3: Input Basic Salary  Step 4: DA = 70% of Basic Salary  HRA = 7% of Basic Salary  MA = 2% of Basic Salary  TA = 4% of Basic Salary  PF = 12% of Basic Salary  IT = 500  Allowances  DA + HRA + MA + TA  Deduction  PF + IT  Net Salary  Basic Salary + Allowances - Deduction  Step 5: Print Net Salary  Step 6: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you learned about various data types that can be suitably used for this problem? Do mention which data types can be used and why? Also mention the difference between the outputs   **Answer:**  In this program two datatypes can be used, int or float. If we use the datatype int then we get the decimal output value (and if the output will be floating value and  we used int datatype then it approximates the floating value and output will be an integer) and if we use float datatype the it will give the floating value as output. (It gives by default output of 6 digits after point) Hence, after calculation process if the answer is floating value and we use integer datatype then we get wrong answer. So, by using float datatype we get more precise answer. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1** | Write a program that takes the length of the pendulum as input and then calculate the time period of the pendulum. Provided that, T=2π√L/G. Define the value of π as 3.14 and take L as the length of the pendulum and G as the acceleration of gravity either in m/s or as input from the keyboard. Display the time period rounded to 2 decimal places.  **Hint:**  Use **Math.h** header file, use #define for specifying the value of π  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Fill up the output as per the inputs mentioned in below given table as per the output received in console.   |  |  |  |  | | --- | --- | --- | --- | | **Sr. No.** | **Input** | **Output** |  | | Length | Gravity | Time Calculated(seconds) |  | | 1. | 50 m | 9.8 m/s2 | 14.19 | | 2. | 50 m | 0 m/s2 | 1.#J | | 3. | 50 m | 0.9993 g | 44.42 | | 4. | 50 m | -1 g | -1.#J | |
| **Flowchart** |  |
| **Algorithm** | Step-1 : Start.  Step-2 : Defining the value ‘pi 3.14’ using #define.  Step-3 : Declare the variables ‘L’ , ‘G’ and ‘T’.  Step-4 : Print “Enter the value of length of pendulum(L).”.  Step-5 : Taking input from user for value of length of pendulum(L).  Step-6 : Print "Enter the value of acceleration of gravity(G).”.  Step-7 : Taking input from user for value of acceleration of gravity(G).  Step-8 : Calculating time period of pendulum(T) using formula -> T = 2\*pi\*sqrt(L/G).  Step-9 : Print ‘T’.  Step-10 : End. |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you learned about, how math function is useful for calculating square root?Which datatype is supported by all math functions? Also mention any 5 mathfunctions with their purpose.   |  |  |  | | --- | --- | --- | | Sr no. | Math function | Description | | 1 | sqrt() | Calculates the square root of a floating point value | | 2 | pow() | Returns the result of a floating-point value raised to a certain power | | 3 | abs() | Returns the absolute value (positive value) of an integer | | 4 | floor() | Rounds up a floating-point value to the next whole No.(nonfractional) value | | 5 | ceil() | Rounds down a floating-point value to the next whole number | |

|  |  |
| --- | --- |
| **3.2(a)** | Let us understand the working of Pre-increment, Post-increment, Pre-decrement and Post-decrement a) Consider a scenario where, Boys are playing in the park and collecting and removing the yellow balls in/from the bucket based on teacher’s instruction. Let’s say there are already 10 Yellow balls present in a bucket. Following is the sequence of the instructions given by the teacher for adding/removing the balls.   1. Rajiv: ++ Yellow 2. Preet: --Yellow 3. Raj: Yellow++   iv. Ritul: Yellow-- |
| **3.2(b)** | Consider another scenario where boys and girls both are asked to add/remove Yellow and Pink balls from the bucket respectively. Currently there are 10 Yellow balls in the bucket and 20 Pink balls. Teacher has given the sequence of instructions as below for adding/removing the balls. Calculate = ++Yellow + Yellow++ + --Yellow + ++Pink - --Pink - --Pink Get the count of Yellow and Pink balls after evaluating above given scenario. |
| **Flowchart(a)** |  |
| **Flowchart(b)** |  |
| **Algorithm (a)** | Step1: start  Step2: declare yellow=10  Step3: print count before execution  Step4: calculations for no. of balls rajiv,preet,raj,ritul has  Step5: print count after execution  Step6 : stop |
| **Algorithm (b)** | Step1: start  Step2: declare yellow, pink, calculate  Step3: assign values to yellow and pink  Step4:print count before execution  Step5:calculate ++yellow + yellow++ + --yellow + ++pink - \_\_pink - --pink  Step6:print count after execution  Step7:stop |
| **Code (a)** |  |
| **Code (b)** |  |
| **Output(a)** |  |
| **Output(b)** |  |
| **Question** | Have you understood the working of Pre-increment, Post-increment, Pre-decrement and Post-decrement?  Ans. : Yes, I understood the working of Pre-increment, Post-increment, Pre-decrement and Post-decrement with these programs. |

|  |  |
| --- | --- |
| **3.3** | Write a C program to swap two numbers (use two variables for collecting value from user) without using third variable. (Hint: Use arithmetic operators) |
| **Flowchart** |  |
| **Algorithm** | Step1: start  Step2: declare a & b  Step3: print and scan the values of number 1 and number 2  Step4: write mathematical process a=a+b , b=a-b , a=a-b  Step5: print before swapping and after swapping  Step6: stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you learned about, how we can use arithmetic operators for swapping the numbers?   Ans. : Yes, now I understood that how we can use arithmetic operators for swapping the numbers |

|  |  |
| --- | --- |
| **4.1(a)** | a. Write something about your characteristics not more than 50 words using gets function and print out the same using puts function. |
| **Flowchart** |  |
| **Algorithm** | Step1:start  Step2:declare array  Step3: print input  Step4: gets()  Step5: print output  Step6: puts()  Step7: end |
| **Code** |  |
| **Output** |  |
| **Question** | What is the significance of using gets and puts? Are they acting as replacement of any function? How?  Ans) gets():reads characters from the standard input and stores them as string.  Puts():prints characters from standard output. just like printf statement |

|  |  |
| --- | --- |
| **4.1(b)** | Write a program to convert the decimal number into octal and hexadecimal format. Print hexadecimal and octal values for given inputs in expected outcomes. |
| **Flowchart** | Diagram  Description automatically generated |
| **Algorithm** | Step1:start  Step2: declare a,b,c,d  Step3: take input of a,b,c,d from user  Step4: print a,b,c,d using %d,%o,%x in integer,octal,hexadecimal form respectievely  Step5:end |
| **Code** |  |
| **Output** |  |
| **Question** | |  |  |  |  | | --- | --- | --- | --- | | Sr.no. | input | octal | hexadecimal | | 1. | 80 | 120 | 50 | | 2. | 143 | 217 | 8f | | 3. | 0 | 0 | 0 | | 4. | 1 | 1 | 1 | | 5. | -1 | 37777777777 | ffffffff | |

|  |  |
| --- | --- |
| **4.2** | Write a C Program to Print multiplication table from 1 to 7 to achieve the following output. (Use #define directives and do while loop) |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start.  Step 2: Declare variable row and col then enters the do-while loop.  Step 3: It enters the inner do-while loop.  Step 4: Execute a group of statement within the inner do-while loop like row\*col.  Step 5: Next we use increment operator inside the loop operator to increment the value.  Step 6: Next it checks the while condition. If the condition is true then the code inside the do-while loop both outer and inner executes again. The process will last until the condition fails.  Step 7: If it is false compiler exits from the loop.  Step 8: Display result  Step 9 :Stop. |
| **Code** |  |
| **Output** |  |

|  |  |
| --- | --- |
| **5.1** | Write a C program for the given scenario from the flowchart. Note that you have to enter your own height in centimeters. |
| **Flowchart** |  |
| **Algorithm** | Step 1 :Start  Step 2 :Input height  Step 3:If height less than 150 print Dwarf height  Else if height between 150 and 165 print Average height.  Else if height between 165 and 195 print Tall height.  Else print abnormal height.  Step 4:End |
| **Code** |  |
| **Output** |  |
| **Question** | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Sr.No.** | **Inputs** | **Dwarf** | **Average** | **Tall** | **Abnormal** | | **Your height** | **174** |  |  | **Yes** |  | | **Mother’S height** | **145** | **Yes** |  |  |  | | **Father’s height** | **160** |  | **Yes** |  |  | | **Sibling’s height** | **179** |  |  | **Yes** |  | |

|  |  |
| --- | --- |
| **5.2** | Write a C program to find all roots of a Quadratic equation using nested switch case. Take three user inputs from keyboard for finding the discriminant (b2 – 4ac). Use the concept of nested switch case for finding the roots of equation. Get the outputs for roots till 2 decimal points only. |
| **Flowchart** |  |
| **Algorithm** | Step1 :Start  Step2 :Input three numbers a,b,c  Step3 :D=(b\*b)-4ac  Step4 :If D>0 print root1 = (-b+sqrt(D))/2a  Root2 = (-b-sqrt(D))/2a  Step5: If D=0 print root1 = root2 = -b / (2\*a)  Step6 :If D<0 print img= sqrt (-D) / (2\*a)  Step7 :End. |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you learned about how to use normal switch case and nested switch case? 2. Is default case necessary for every switch case?  3. What if break statement is not mentioned between two consecutive cases?  Ans1) yes I have successfully learned how to use switch case  Ans2)no it is not necessary to use default case.  Ans3)if break statement is not mentioned then it will run all the case even if the first case matches the condition. |

|  |  |
| --- | --- |
| **5.3** | If the ages of Ram, Shyam and Ajay are input through the keyboard, write a program to determine the youngest of the three. If all of them are of same age then print that “All are of same age”. (Hint: Use Nested if else statement) |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare Ajay, Shyam , Ram  Step 3:Input Age of Ajay ,Shyam ,Ram  Step 4: if Ram<Shyam go to step 6 else go to step 5  Step 5: if Shyam<Ajay go to step 8 else go to step 9  Step 6: if Ram<Ajay go to step 7 else go to step 9  Step 7: Ram Is Youngest  Step 8:Shyam Is Youngest  Step 9: Ajay Is Youngest  Step 10: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you tried merging the concepts of Nested if else and else if ladder in this scenario?  Ans1)yes, I have tried to use nested if else and else if ladder in the scenario |

|  |  |
| --- | --- |
| **5.4** | The policy followed by a company to process customer orders is given by the following rules: Suppose stock=100  a) If a customer order is less than or equal to that in stock and ‘has credit’ is OK, supply ‘has requirements.  b) If ‘has credit’ is not OK do not supply. Send him intimation.  c) If ‘has credit’ is OK but the item in stock is less than ‘has ordered’, inform ‘out of stock’ and intimate him that the balance will be refunded.  Write a C program to implement the company policy. |
| **Flowchart** |  |
| **Algorithm** | Step1: start  Step2: display do you have credit.  Step3: if yes and stock>=order  Stock=stock -order  Else if stock < order  Out of stock  Step4: if no credit then display order rejected  Step5: stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Which kind of logic have you used for building this program? If else if ladder or nested if else statements?   Ans1) else if ladder. |

|  |  |
| --- | --- |
| **6.1** | There is a person, who is asked to enter the alphanumeric password for registering into an ecommerce website for purchasing products from website. But he is not aware about, what does Alphanumeric mean. So, he tries entering various combinations 5 times, but he fails to create such password. So let us help him by writing a C program to validate his password. Constraints for writing password are it should have combination of lowercase, uppercase and digit. Note: Use Do while loop, and give print appropriate outputs on incorrect validations. |
| **Flowchart** |  |
| **Algorithm** | Step1 : start  Step2 : declare array and declare length, digit, i , upper, lower  Step3 : print enter your password  Step4 : use for loop to continue to check the length of the password  Step5: now check if digit, upper case, lower case are present or not using for loop and if condition  Step6 : print password is good if condition matches else print password should contain alphanumeric values.  Step7 : stop. |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Have you understood working of do…while loop? Do mention the syntax of this loop.  2. Have you used for loop in this program?  Ans1)ye I have understood do…while loop  Syntax:-  Do  {  Initialization;  Statement;  Modification;  }while(condition)  Ans2) yes I have used for loop in this program. |

|  |  |
| --- | --- |
| **6.2** | Two numbers are entered through the keyboard. Write a program to find the value of one number raised to the power of another. (Use While loop) |
| **Flowchart** |  |
| **Algorithm** | Step1: start  Step2: declare num,pow,ans  Step4:print base value and power  Step5: take input from user  Step6: use while loop to get the output  Step7: print the ans  Step8: stop |
| **Code** |  |
| **Output** |  |
| **Question** | Questions: 1. Have you understood the concept of while loop? if yes write its syntax here.  Ans1) yes I have understood the concept of while loop.  Syntax:  initialization  While(condition)  {  modification  } |

|  |  |
| --- | --- |
| **6.3** | Write a C program for Big bazaar cashier to count the amount to be collected from the customer. Cashier will enter the numbers one after another for each item and to get the summation of entered numbers, he has to enter 0. (Use for loop) (Hint: Break statement can be used) |
| **Flowchart** |  |
| **Algorithm** | Step1: start  Step 2: declare total items ,item number, rate, number of items.  Step3: use for loop  Step4: if condition is true print item number  Step5: use if condition  Step6: if condition is true then proceed if false then stop.  Step7: write arithmetic operations and print total price  Step8: stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1)Have you learned the concept of for loop using above given scenario? Explain what does ‘i’ stands for in the for() loop, consider the given example below. E.g. for(i=0;i<10;i++)  Ans) yes I understood the concept of loop.  Here, I in for loop indicates the number of time the loop will execute. |

|  |  |  |
| --- | --- | --- |
| **6.4** | Write a program for a match-stick game between the computer and a user. Your Program should ensure that the computer always wins. Rules for the games are as follows: ● There are 21 match-sticks. ● The computer asks the player to pick 1, 2, 3, or 4 match-sticks. ● After the person picks, the computer does its picking. ● Whoever is forced to pick up the last match-stick loses the game. Use while loop, break and Continue Statements. | |
| **Flowchart** |  | |
| **Algorithm** | Step1: start  Step2: declare num,user,comp  Step3: while num>1 print user turn  Step4: use if condition  Step5: write arithmetic operations  Step6:print computer’s turn  Step7: if (condition) true then proceed else goto step8  Step8: stop | |
| **Code** |  | |
| **Output** |  | |
| **Question** | 1. What is the significance of using break and continue statement?   Ans) significance of break statement: break statement is used to terminate the loop  Significance of continue: continue is used to bring the program control to the beginning of the loop. | |
| **7.1** | | Twenty-five numbers are entered from the keyboard into an array. Write a C program to find out how many numbers of them are positive, negative, and how many are even and odd?  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the counts of positive, negative, even and odd numbers in the below given table as per the output received. |
| **Flowchart** | |  |
| **Algorithm** | | **Algorithm:**  Step 1: Start  Step 2: Declare variables i, b=0,c=0,d=0,e=0 and array a[25]  Step 3: for i<2  if a[i]<0 go to step 4 else go to step 5 Step 4: ++c  Step 5: ++bStep 6: if a[i]%2==0 go to step 7 else go to step 8 Step 7: ++d  Step 8: ++e  Step 9: Print positive, negative, odd &even numbers Step 10:Stop |
| **Code** | |  |
| **Output** | |  |
| **Question** | | **Is it necessary to initialize a variable with zero every time? If yes, then why? If**  **No, then when is it necessary to initialize the number with zero and why?**  Ans: Yes, it is necessary to initialize a variable with zero every time because if we don’t initialize system would take a garbage value for that variable. |

|  |  |
| --- | --- |
| **7.2** | Write a program for creating two arrays of different size and merge both arrays into one by sorting those arrays in ascending order. [**Merge by sorting**]  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Following screenshot showcases the expected outcome, you can enter the input values of your choice |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare variables n1,n2,n3,i,j,temp Step 3: Take input of array sizes from the user Step 4: n3=n1+n2  Step 5: Declare array a[n1],b[n2],c[n3].  Step 6: For i<n1 if condition is true Take input of first array elements and store it in third array else go to step 7  Step 7: For i<n2 if condition is true Take input of first array elements and store it in third array else go to step 8  Step 8: for j=i+1,i<n3,j++ if condition is true go to step 9 else go to step 11 Step 9:If c[i]>c[j] if condition is true go to step 10 else go to Step 8  Step 10: temp=c[i]  C[i]=c[j]  C[j]=temp  Step 11:Print the sorted array  Step 12:Stop |
| **Code** | #include<stdio.h>  void main()  {  int i,k,j,n1,n2,n3,a[50],b[50],c[50];  printf("Enter the size of a : ");  scanf("%d",&n1);  for(i=0;i<n1;i++)  {  printf("enter a[%d] : ",i);  scanf("%d",&a[i]);  c[i]=a[i];  k=i;  }  printf("Enter the size of b : ");  scanf("%d",&n2);  n3=n1+n2;  for(i=0;i<n2;i++)  {  printf("enter b[%d] : ",i);  scanf("%d",&b[i]);  c[n1]=b[i];  n1++;  }  int h;  for(k=0;k<n3;++k)  {  for(j=k+1;j<n3;j++)  {  if(c[k]>c[j])  {  h=c[k];  c[k]=c[j];  c[j]=h;  }  }  }  printf("\nmerge and short of array 'a' and 'b' \n");  for(i=0;i<n3;i++)  {  printf("%d",c[i]);  }  } |
| **Output** |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **7.3** | Write a Program to multiply any two 3\*3 Matrices.  **Test Data:**  Input the rows and columns of first matrix: 3 3  Input the rows and columns of second matrix: 3 3  **Expected Input and Output:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Input for first matrix:   |  |  |  |  | | --- | --- | --- | --- | |  | J[0] | J[1] | J[2] | | I[0] |  |  |  | | I[1] |  |  |  | | I[2] |  |  |  |   Input for Second Matrix:   |  |  |  |  | | --- | --- | --- | --- | |  | J[0] | J[1] | J[2] | | I[0] |  |  |  | | I[1] |  |  |  | | I[2] |  |  |  | |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare variables mat1[3][3], mat2[3][3], mul [3][3], i, j, k Step 3: for i<3 if condition is true go to step 4 else go to step 6  Step 4: for j<3,j++ take input of rows and columns else go to step 5 Step 5: i++  Step 6: for i<3 if condition is true go to step 7 else go to step 9 Step 7: for j<3,j++ take input of rows and columns else go to step 8 Step 8:i++  Step 9: for i<3 if condition is true go to step 10 else go to 14  Step 10: for j<3 if condition is true go to Step 11 else go to step 13  Step 11:for k<3 if condition is true mul[i][j]+=mat1[i][k]\*mat2[k][j] else go to step 12  Step 12 ;j++  Step 13 :i++  Step 14 :printf |
| **Code** |  |
| **Output** |  |
| **Question** | **State the advantages of using Array Indexes. When is it suitable to take array index?**  Ans: In arrays, the elements can be accessed randomly by using the index number. Arrays allocate memory in contiguous memory locations for all its elements. Hence there is no chance of extra memory being allocated in case of arrays. This avoids memory overflow or shortage of memory in arrays. |

|  |  |
| --- | --- |
| **8.1** | Help user to identify how strong is his password based on the number of lowercase alphabets, uppercase alphabets, digits and special characters given by the user from the keyboard. Length of entered password(string) should be of 8.  **Constraints for identifying strength of password:**  **1. Strong:** Mixture of lowercase alphabets, uppercase alphabets, digits and special characters  **2. Average:** Mixture of lowercase alphabets, digits and special characters  **3. Poor:** Either only has alphabets/digits/special characters  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Look at the example given in table and try various test cases such a way to get the varied strength of password (Strong/Average/Poor). |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2 : Declare variablesu=0,l=0,s=0,n=0,j=0,password[8];  Step 3 : enter the password.  Step 4 :i=0 check password[i]!=0 if no than check isuppre if yes than u=u+1  Step 5 : check islower else if yes than l=l+1. check isdigit else if yes than d=d+1.else s=s+1.  Step 6 :check all condition and check incriment  Step 7 : if u>0 && l>0 && n>0 && s>0 than print strong.  Step 8 : else if l>0 && n>0 && s>0than print average.  Step 9 : else if s>0 || l>0 || n>0 || u>0than print week.  Step 10 :stop |
| **Code** |  |
| **Output** |  |
| **Question** | Explain the difference between string and character. Also write the syntax for printing character and string.  Ans : The main difference between Character and String is that Character refers to a single letter, number, space, punctuation mark or a symbol that can be represented using a computer while String refers to a set of characters. In C programming, we can use char data type to store both character and string values.  A string is a sequence of characters terminated with a null character \0 ........................................................................................................................................ For example : char c[] = "c string";  A character is terminated with a null character \0 . ...  For example: char c[] = ‘c’; |

|  |  |
| --- | --- |
| **8.2** | Let us assume, teacher is supposed to allot seats based on the student’s names. You are requested to help teacher by creating a C program, for collecting the names of 5 students and sort them in alphabetical order.  **Hint:** Use string functions, use **fgets** function to collect the names of students.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the inputs entered by you for 5 names, and give the output how they are sorted. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2 : Declare variables I ,j , count, str[25][25] , temp[25]  Step 3 : enter the number of student  Step 4 :i=0 check i<=count if yes than get name of student  Step 5 :i=1 else if check i<=count yes then j=i+1 check j<=count .  Step 6 :check strcmp(str[i],str[j])>0 if yes then swap else i=i+1.  Step 7 :i=0, check i<=count yes then print string  Step 8 :Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Which string functions have you learned from this program? Explain any 5 string functions in below given table.   |  |  |  | | --- | --- | --- | | **Sr. No.** | **Input of names** | **Sorted Output as per output** | | 1. | Strlen(a) | To find string length | | 2. | Strcpy(a) | Copy string to another string | | 3. | Strcat(a) | Concatenation of string | | 4. | Strcmp(a) | To compering two string | | 5. | Strrev(a) | To reverse string | |

|  |  |
| --- | --- |
| **8.3** | Write a C program to check if the user inputted string is palindrome or not using recursion.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the following test inputs and give the output as per the output gained. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare variablesI, l, n, j=0 , a[50].  Step 3: Read the string and find the length.  Step 4: i=0. if i<l/2 if yes than check a[i]==a[l-1-i] || a[i]==32+a[l-1-i] || a[i]==a[l-1-i]-32 Step 5: j=j+1, i=i+1  Step 6: j=0, j>=n  Step 7: if yes than palindrome else if j<n then not palindrome  Step 8: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Explain the concept of recursion. Explain the difference between recursion and iteration?  >The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called as recursive function.   |  |  | | --- | --- | | RECURSION | ITERATION | | The statement in a body of function calls the function itself | Allows the set of instructions to be repeatedly executed. | | Recursion reduces the size of the code. | Iteration makes the code longer. | |

|  |  |
| --- | --- |
| **9.1** | Write a C program to check if the entered number is prime or not by using types of user defined functions  (i) No arguments passed and no return value  (ii) No arguments passed but a return value  (iii) Argument passed but no return value  (iv) Argument passed and a return value  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the details into the table based on the inputs entered by you and tick mark the column, whether the inputted value is prime or non-prime: |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Prime() Step 2a: Input a number.  Step 2b:IF i<=n THEN GOTO Step 2c  Step 2c:IF i%2==0 THEN Print prime  ELSE print not prime  ENDIF  Step 2d:i++  Step 3:Stop |
| **Code(i)** |  |
| **Code(ii)** |  |
| **Code(iii)** |  |
| **Code(iv)** |  |
| **Output** |  |
| **Question** | 1. You might be clear now, how user defined functions are created in different ways. Explain them.  > A User-defined functions are those functions which are defined by the user at the time of writing program. These functions are made for code reusability and for saving time and space. |

|  |  |
| --- | --- |
| **9.2** | Verify the triangle, if the length of the sides of a triangle are denoted by a, b and c, then the area of triangle is given by:    Use nested function.  Collect the values for a, b and c from user for identifying whether it forms Triangle or not.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the inputs for verifying triangle and mention the results in the below mentioned table format. Tick mark whether based on input, triangle is formed or not. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Sum() Step 2a: Input a,b,c,c,s  Step 2b: s=(a+b+c)/2 area1=area(a,b,c,s)  Step 2b1: Input a,b,c,s  Step 2b2: area0=sqrt(s \*(s-a)\*(s-b)\*(s-c))  Step 2c: IF a+b>c && b+c>a && c+a>b  THEN Print valid triangle  Print area  ELSE Print invalid triangle  Step 3:Stop |
| **Code** |  |
| **Output** |  |
| **Question** | **Explain the concept of nested functions in C.**  Ans: A nested function is a function defined inside another function. Nested functions are supported as an extension in GNU C, but are not supported by GNU C++. The nested function's name is local to the block where it is defined. |

|  |  |
| --- | --- |
| **9.3** | A positive integer is entered through the keyboard, write a function to find the binary equivalent of this number using recursion.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the inputs for converting the number into binary form, try it for three different inputs and fill the below given table: |
| **Flowchart** |  |
| **Algorithm** | Step 1:Start  Step 2:bin(n) Step 2a: bin(int n)  Step 2b: IF n>0 THEN bin(n/2)  Print n%2  Step 3: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Mention the advantages of using recursion in a program.  • Recursion can reduce time complexity. ...  • Recursion adds clarity and reduces the time needed to write and debug code. |

|  |  |
| --- | --- |
| **10.1** | Write a C program to create a structure of Book Detail and display the details of the book in appropriate format by passing structure as a function argument.  Book Detail must contain following information:  Book Title, Author name and Amount of book in float.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the inputs for converting the number into binary form, try it for three different inputs and fill the below given table: |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare a structure (book) which consists details of book  (With Membrers -name, author , price)  Step 3: Take input from user(Name of book, author, price)  Step 4: Print book details  Step 5:Stop |
| **Code** |  |
| **Output** |  |
| **Question** | **Can we declare function inside structure of C Programming? Explain Why?**  Ans: No, you cannot define a function inside the structure of C Programming, but you can do so in C++, rather you can have a function pointer in a “struct” in C Language. |

|  |  |
| --- | --- |
| **10.2** | Create a **Union** called library to hold accession number, title of the book, author name, price of the book and flag indicating whether the book is issued or not. (flag = 1 if the book is issued, flag = 0 otherwise). Write a program to enter data of one book and display the data.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the inputs for collecting the details for library books. Here, if user inputs flag=1, then book is issued else book is not issued. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare a union with members holding character value ,float and integer value.  Step 3: Declare union variables a1,a2,f.  Step 4: Take input from the user and store it in union variables.member according  to requirement.  Step 5: Check the flag using ‘ if ’ condition(if flag is equal to 1 then book is issued).  Step 6: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | **Explain the major difference between structure and union in detail.**  Ans: A structure is a user-defined data type available in C that allows to combining data items of different kinds. Structures are used to represent a record. A union is a special data type available in C that allows storing different data types in the same memory location. |

|  |  |
| --- | --- |
| **10.3** | Write a C program for collecting and displaying employee details such as, Age, Name, Address and Salary by using nested structure.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare a structure (employee) which consists details of employee  (With Membrers -name, address , salary, age)  Step 3: Take input from user(Name of employee, address , salary, age )  Step 4: Print employee details  Step 5:Stop |
| **Code** |  |
| **Output** |  |
| **Question** | **Explain how nested structure works in C programming.**  Ans: A nested structure in C is a structure within structure. One structure can be declared inside another structure in the same way structure members are declared inside a structure. |

|  |  |
| --- | --- |
| **11.1** | Write a program to read the marks of 10 students for the subject CE143 Computer concepts and Programming and computes the number of students in categories FAIL, PASS, FIRST  CLASS and DISTINCTION using Pointers and Arrays. |
| **Flowchart** |  |
| **Algorithm** | Step 1 : Start  Step 2 : Input marks  Step 3: IF marks>70 THEN  ++w and add count into Distinction  ELSE GOTO STEP 4  Step 4: IF marks> 60 and marks<70 THEN  ++x and add count into FIRST CLASS  ELSE GOTO STEP 5  Step 5: IF marks> 40 and marks<59 THEN  ++y and add count into PASS  ELSE  ++z and add count into FAIL  Step 6: Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Explain the importance of using pointers?  1. Pointers provide direct access to memory  2. Pointers provide a way to return more than one value to the functions  3. Reduces the storage space and complexity of the program  4. Reduces the execution time of the program |

|  |  |
| --- | --- |
| **11.2** | Write output for the following programs: |
| **1.** | (Pointers to Functions)  #include<stdio.h>  void display();  int main()  {  void (\*func\_ptr)();  func\_ptr=display;  printf("Address of functions display is %u\n",func\_ptr);  (\*func\_ptr)();  return 0;  }  void display()  {  puts("By helping others, we help overselves!!");  } |
| **output** |  |
| **2.** | (Functions Returning Pointers) char \*copy (char\*,char \*); int main()  {  char \*str;  char source[] = "Kindness"; char target[10]; str=copy(target,source); printf("%s\n",str);  return 0;  }  char \*copy(char \*t,char \*s)  {  char \* r; r = t;  while(\*s!='\0')  {  \*t=\*s; t++; s++;  }  \*t='\0'; return(r);  } |
| **Output** |  |

|  |  |
| --- | --- |
| **12.1** | Write a program to read a text file ‘Demo.txt’ and print each word of that file in reverse order. |
| **Flowchart** |  |
| **Algorithm** | Step 1 : Start  Step 2 : Declare file pointer fp, character ch, and integer i and pos  Step 3 : Open file Demo.txt  Step 4 : If f(p==NULL) then print filr does not exist..!!  Step 5 : Else fseek(fp,0,SEEK\_END)  Step 6 : i=0  Step 7 : If (i<pos)  i++ and fseek(fp,-I,SEEK\_END) then print output  Step 8 : End |
| **Code** |  |
| **Output** |  |
| **Question** | **Explain, why do we need to use files in C?**  Ans. Entire data is lost when the program terminates and storing in a file will preserve your data even if the program terminates. If you want to enter a large amount of data, normally, it takes a lot of time to enter them all.  If you have a file containing all the data, you can easily access the contents of the file by using few commands in C. you can easily move  your data from one computer to another without changes. |

|  |  |
| --- | --- |
| **12.2** | Write a C program that illustrates how to write into a file using **putw()** function and how to read the same file using **getw()** function.  Use **fopen(), fclose(), getw() and putw()** functions. |
| **Flowchart** |  |
| **Algorithm** | Step 1 : Start  Step 2 : Declare file pointer fp, and integer n1 and n2  Step 3 : Open file num.txt  Step 4 : If (fp==NULL)  Then print file does not exist..!!  Step 5 : Else print enter an integer value  Step 6 : get value in n1  Step 7 : put n1 in fp  Step 8 : close file fp  Step 9 : Open file num.txt  Step 10 : get fp in n2  Step 11 : print n2  Step 12 : close file fp  Step 13 : End |
| **Code** |  |
| **Output** |  |
| **Question** | **1.Explain any 3 functions of file other than mentioned in the problem.**   |  |  |  | | --- | --- | --- | | **Sr. No** | **Function** | **Purpose** | | **1.** | feof() | Fubctin finds the end of file. | | **2.** | fscanf() | Function reads formatted data from file. | | **3.** | fprintf() | Function writes formatted data to a file. | |

|  |  |
| --- | --- |
| **12.3** | Two files Data1.txt and Data2.txt contains list of integers. Write a program to produce  file Data3.txt which holds as merged list of these two lists.  Use **command line argument** to specify the file name. |
| **Flowchart** |  |
| **Algorithm** | Step 1 : Start  Step 2 : Declare file pointer fp1,fp2,fpmerge then  character fname1[20],fmane2[20],fname3[30]  Step3 : Enter the first file and second file name  Step 4 : Enter your new file name  Step 5 : pen file fname1,fname2,fname3 in fp1,fp2,fpmerge  Step 6 : If (ch=fgetc(fp1)!=EOF)  Then fputc(ch,fpmerge)  Step 7 : Else if (ch=fgetc(fp2)!=EOF)  Then fputc(ch,fpmerge)  Step 8 : Print The two file merged into new file successfully..!!  Step 9 : Close file fp1,fp2,fpmerge  Step 10 : End |
| **Code** |  |
| **Output** |  |
| **Question** | **Explain the difference between argc and argv along with their significance.**  **1.** Command line arguments are passed to the main() method. Here argc counts the number of arguments on the command line and argv[ ] is a pointer array which holds pointers of type char which points to the arguments passed to the program.  **2.** argc is the number of arguments being passed into your program from the command line and argv is the array of arguments. |

|  |  |
| --- | --- |
| **13.1** | Twenty-five numbers are entered from the keyboard into an array. Write a C program to find out how many numbers of them are positive, negative, and how many are even and odd?  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the counts of positive, negative, even and odd numbers in the below given table as per the output received. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Declare a structure studen) which consists details ofstudent  (With Membrers -name, roll , age , branch, class)  Step 3: Take input from user(Name, roll , age , branch, class)  Step 4: Print student details  Step 5:Stop |
| **Code** |  |
| **Output** |  |
| **Question** | 1. Explain the benefits of using dynamic memory allocation. Give one scenario where it is most useful.  • When we do not know how much amount of memory would be needed for the program beforehand.  • When we want data structures without any upper limit of memory space. |

|  |  |
| --- | --- |
| **13.2** | Twenty-five numbers are entered from the keyboard into an array. Write a C program to find out how many numbers of them are positive, negative, and how many are even and odd?  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the counts of positive, negative, even and odd numbers in the below given table as per the output received. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Input char ptr  Step 3:ptr=(char)malloc(10)  Step 4: copy the string using strcpy(ptr, "programming")  Print ptand address of ptr  Step 5: ptr = (char\*)realloc(ptr , 20)  Step 6: add C in ptr using strcat(ptr,"in 'c'")  Print ptr and address of ptr  Step 6: free(ptr) to free remaining memory  Step 7:Stop |
| **Code** |  |
| **Output** |  |
| **Question** | **1. Mention advantage of using realloc() function.**  **>** It increases or decreases the size of the specified block of memory, moving it if necessary |

|  |  |
| --- | --- |
| **13.3** | Twenty-five numbers are entered from the keyboard into an array. Write a C program to find out how many numbers of them are positive, negative, and how many are even and odd?  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the counts of positive, negative, even and odd numbers in the below given table as per the output received. |
| **Flowchart** |  |
| **Algorithm** | Step 1: Start  Step 2: Input array elements  Step 3: ptr = (int\*)malloc(limit \* sizeof(int))  Step 4: IF i<limit THEN  Input ptr+i  Print ptr+i  sum += \*(ptr + i)  avg=sum/limit  Step 5: Print avg  Step 6: free(ptr)  Step 7: Stop |
| **Code** |  |
| **Output** |  |