**CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY**

**DEVANG PATEL INSTITUTE OF ADVANCE TECHNOLOGY & RESEARCH**

Department of Computer Engineering/Computer Science & Engineering/

Information Technology

**Subject Code and Name:CE143 Computer Concepts and Programming**

**Semeter:1**

**Academic year:2021-22**

|  |  |
| --- | --- |
| **No.** | Aim of the Practical |
| **Set – 1** | |
| **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.  **Outcome:**    Note:  There are three shapes in the output: Smiling Face, Diamond & Heart.  The ASCII Value for Smiling face is 1.  The ASCII Value for Diamond is is 4.  The ASCII Value for Heart is is 3.  Also draw flowchart and write algorithm.  Try this example on Turbo C or Code blocks only.  **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**  **Flowchart:**    **Code**  #include<stdio.h>  #include<conio.h>  void main()  {  clrscr();  printf("%c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c\n",1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1);  printf("%c\"If you are resisting somethig,you are feeling it.\t\t\t%c\n",4,4);  printf("%c\tAny energy you fight,you are feeling it\t\t\t\t%c\n",3,3);  printf("%c\t\tif you are pushing something away\t\t\t%c\n",1,1);  printf("%c\t\t\tYou are inciting it to stay.\"by Micheal Singer.\t%c\n",4,4);  printf("%c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c\n",1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1);  getch();  }  **Output Screenshot**    **Question- Answer**  **1. Have you learnt about ASCII values for different symbols other than smile,**  **diamond and heart? If yes, then mention any 5 ASCII symbols and their values in**  **tabular format.**    **Sr. No. Symbol ASCII Value**  **All the content should be in Times New Roman, size 12 and line spacing 1.15.** |
| **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.  Expected Outcome:  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot  of output.    **Algorithm:**  **Step 1: Start.**  **Step2: Print Your Bio Data as per given in question.**  **Step3: Stop.**  **Flow Chart:**    **Code:**  #include<stdio.h>  #include<conio.h>  void main()  **{**  printf("#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#");      printf("\n \t\t\t\t\tbio - data");      printf("\n#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#");   printf("\n\n \t\t\t\tbasic information");  printf("\n \t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");        printf("\n \t\t\t\tname            :%s   ",name);      printf("\n \t\t\t\taddress         :%s   ",adddress);      printf("\n \t\t\t\tmobile number   :%s   ",mob);      printf("\n \t\t\t\tdate of birth   :%s   ",dob);      printf("\n \t\t\t\tgender          :%s   ",gender);      printf("\n\n \t\t\t\t  education qualification");      printf("\n \t\t\t\t  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");      printf("\n ssc-> name of school   %s    passing year  %s    percentage %s",nos1,py1,percen1);      printf("\n ssc-> name of school   %s    passing year  %s    percentage %s",nos2,py2,percen2);      printf("\n \t\t\ttechnical skills :%s",ts);      printf("\n \t\t\thobies           :%s",hobbies);  printf("\n#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#");      printf("\n \t\t\t\t\t thank you ");      printf("\n#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#=======#");  getch();  }  Output Screen shot:    **Questions:**   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 Sequence | Purpose | | 1 | \t | Horizontal Tab | | 2 | \n | New Line | | 3 | \a | Back alert | | 4 | \f | Form feed | | 5 | \\ | Back slash | |

|  |  |
| --- | --- |
| No. | Aim of the Practical |
| Set-2 | |
| 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 |  |   **Algorithm:**  **Step 1:** Start  **Step2**: Declare all variable and pop = 80,000  **Step3**: Display population = 80000  **Step4:** Number of Literate = pop\*0.48  **Step5:** Number of Men = pop\*0.52  **Step6:** Number of Literate Men = pop\*0.35  **Step7:** Number of illiterate Men = pop\*0.52-po\*0.35  **Step8:** Number of Women = pop\*0.48  **Step9:** Number of Literate Women = pop\*0.13  **Step10:** Number of illiterate Wimen = pop\*048-pop\*0.13  **Step11:** Printing all details  **Step12**:Stop  **Flowchart:**    **Code:**  #include<stdio.h>  #include<conio.h>  void main()  {  clrscr();  float pop;  pop = 80000;  printf("1\tTotal Population \t%.1f\n",pop);  printf("2\tNumber of Literate \t%.1f\n",pop\*0.48);  printf("3\tNumber of Men \t%.1f\n",pop\*0.52);  printf("4\tNumber of Literate Men \t%.1f\n",pop\*0.35);  printf("5\tNumber of illiterate Men\t%.1f\n",pop\*0.52-pop\*0.35);  printf("6\tNumber of Women \t%.1f\n",pop\*0.48);  printf("7\tNumber of literate Women\t%.1f\n",pop\*0.13);  printf("8\tNumber of illiterate Women\t%.1f\n",pop\*0.48-pop\*0.13);  getch();  }  **Output:**    **Questions:**   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.   Float : stores floating-point values, that is, values that have potential decimal places    Int : only stores integral values, that is, whole numbers |
| 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 | Count | | 1 | Requirement of 100RS.note | 5 | | 2 | Requirement of 150RS.note | 1 | | 3 | Requirement of 10RS.note | 1 |   **Algorithm:**  **Step1; Start**  **Step2; Declare all the variables**  **Step3: Input amount**  **Step4: num\_100not=amount/100**  **Step5: num\_50not=(amount-num\_100note\*100)/50**  **Step6: given\_ammount=(num\_100note\*100)+(num\_50note\*50)**  **Step7:num\_10note=(amount-given\_ammount)/10**  **Step8:Print the total num of 100,50 & 10 notes**  **Step9: Stop**  **Flowchart:**    **Code:**  #include<stdio.h>  #include<conio.h>  void main()  {  clrscr();  int a;  printf("Enter Your Amount\n");  scanf("%d",&a);  printf("1 Requirement of 100Rs.note\t %d\n",a/100);  printf("2 Requirement of 50Rs.note\t %d\n",a/50);  printf("3 Requirement of 10Rs.note\t %d",a/10);  getch();  }  **Output:**    **Questions:**  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 Type | Format Specifier | Example of Data | | 1 | Int | %d | 23 | | 2 | Float | %f | 143.3 | | 3 | Char | %c | v | |
| 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/Output | Amount | | 1 | Enter Your basic Salary | 5000 | | 2 | DA of Basic Salary | 3500 | | 3 | HRA of Basic Salary | 350 | | 4 | MA of Basic Salary | 100 | | 5 | TA of Basic Salary | 200 | | 6 | PF of Basic Salary | 600 | | 7 | Gross Salary | 9150 | | 8 | Net Salary | 8050 |   **Algorithm:**  **Step 1:** Start  **Step 2:** Declaring Variables salary,da,hra,tma,ta,pf,gross,it,net  **Step 3:** DA = salary\*0.7  **Step 4:** HRA= salary\*0.07  **Step 5:** TMA= salary\*0.02  **Step 6 :** TA = salary\*0.04  **Step 7:** PF = salary\*0.12  **Step 8:** Gross = salary+da+hra+tma+ta  **Step 9:** Net = (Salary+da+hra+tma+ta)-pf-it  **Step 10 :** Printing All values  **Step 11 :** Stop  **Flowchart:**    **Code:**  #include<stdio.h>  #include<conio.h>  void main()  {  clrscr();  float salary,da,hra,tma,ta,pf,gross,it,net;  printf("Enter your salary\n");  scanf("%f",&salary);  printf("Enter it value\n");  scanf("%f",&it);  printf("1\tEnter Basic Salary \t %.1f\n",salary);  da = salary\*0.7;  printf("2\tDA of Basic Salary \t %.1f\n",da);  hra = salary\*0.07;  printf("3\tHRA of Basic Salary \t %.1f\n",hra);  tma = salary\*0.02;  printf("4\tMA of Basic Salary \t %.1f\n",tma);  ta = salary\*0.04;  printf("5\tTa of Basic Salary \t %.1f\n",ta);  pf = salary\*0.12;  printf("6\tPF of Basic Salary \t %.1f\n",pf);  gross = salary+da+hra+tma+ta;  printf("7\tGross Salary \t %.1f\n",gross);  net = (salary +da+hra+tma+ta)-pf-it;  printf("8\tNet Salary \t %.1f\n",net);  getch();  }  **Output ScreenShot:**    **Questions: 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.  We use float datatype in this problem.  Float datatype is used because allowance and deduction are to be calculated in % so they might have decimal in it.  If we use integer it might create a big issue in future.  If we use integer the answer we get would be rounded up creating a big issue.  But while using float we get more precise answer |

|  |  |
| --- | --- |
| No. | Aim of Practical |
| Set-3 | |
| 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(second) | | 1 | 50 | 9.8m/s2 | 14.185 | | 2 | 50 | 0m/s2 | Not define | | 3 | 50 | 0.993g | 44.42 | | 4 | 50 | -1g | error |   **Algorithm:**  Step 1:start  Step 2:define pi  Step 3:input length of pendulum & acceleration of gravity  Step 4:calculation of time period of pendulum step 5:print the time period of pendulum  step 6: close  **FlowChart:**    **Code:**  #include<stdio.h>  #include<conio.h>  #include<math.h>  #define pi 3.14  void main()  {  float g,l,t,x;  clrscr();  printf("enter the length of pendulam");  scanf("%f",&l);  printf("enter the value of g as a acceleration");  scanf("%f",&g);  x = l/g;  t = 2\*pi\*sqrt(x);  printf("time period of the pendulam= %.2f",t);  getch();  }  **Output ScreenShot**    Questions:   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 math functions with their purpose.  |  |  |  | | --- | --- | --- | | Sr.No | Math function | Description | | 1 | Sqrt(x) | Square root of x | | 2 | exp(x) | Exponential function | | 3 | Fabs(x) | Absolute value of x | | 4 | pow(x,y) | X raised to power y | | 5 | Ceil(x) | Rounds x to the smallest integer not less than x | |
| 3.2 | 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.  i. Rajiv: ++ Yellow  ii. Preet: --Yellow  iii. Raj: Yellow++  iv. Ritul: Yellow—  Expected Outcome:  Fill up the data mentioned in below given table as per the output received.   |  |  |  | | --- | --- | --- | | Sr.No | Instruction | Yellow | | 1 | Count before execution | 10 | | 2 | Count after execution | 10 |   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.   |  |  |  |  | | --- | --- | --- | --- | | Sr.NO | Instruction | Yellow | Pink | | 1 | Count before execution | 10 | 11 | | 2 | Count After execution | 20 | 19 |   **Algorithm:**  step 1:start  step 2:increase and decrease in ball  step 3: print the all increase & decrease balls  step 4: calculation of the ball  step 5: print the total calculation  step 6: close  **FlowChart:**    **code for a:**  #include<stdio.h>  #include<conio.h>  void main()  {  int yellow=10,cal;  clrscr();  ++yellow;  --yellow;  yellow++;  yellow--;  printf("yellow after execution=%d\n",yellow);  //cal = (++yellow) + (--yellow) + (yellow++) + (yellow--);  //printf("cal=%d",cal);  getch();  }  **code for b:**  #include<stdio.h>  #include<conio.h>  void main()  {  int yellow=10 , pink=20 , cal;  clrscr();  ++yellow;  start  increase and  decrease in ball  & calculation of the  total ball  print the total  calculation &  all increase &  decrease balls  close  yellow++;  --yellow;  ++pink;  --pink;  cal=(++yellow) + (yellow++) + (--yellow) + (++pink) - (--pink) - (--pink);  printf("yellow=%d \n pink=%d\n cal=%d",yellow,pink,cal);  getch();  }  **Output Screen Shot:**      Questions:  Have you understood the working of Pre-increment, Post-increment, Pre-decrement and  Post-decrement?  Rubrics: Output should be as mention  ned in the expected outcome, if it is imperfect then  submission marks are proportional  **Ans.** Yes I have understood the working ok pre increment,post increament,pre decrement,post decrement. |
| 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)  Expected Outcome:  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot  of output.  Fill up the output as per the output received in console   |  |  |  |  | | --- | --- | --- | --- | | Sr.No | Instruction | Number1 | Number2 | | 1 | Before Swapping |  |  | | 2 | After Swapping |  |  |   Algorithm:  algorithm:  step 1: start  step 2: input 2 number by user  step 3: num1 = num1 + num2  num2 = num1 - num2  num1 = num1 - num2  step 4: print the num1 & num2  step 5: close  Flowchart:    Output Screenshot:    Questions:  1. Have you learned about, how we can use arithmetic operators for swapping the  numbers?  **Ans.** yes I understood the working of Swapping two number |

|  |  |  |
| --- | --- | --- |
| Sr.No | | Aim of the Practical |
| Set-4 | | |
| 4.1 | | a. Write something about your characteristics not more than 50 words using gets  function and print out the same using puts function.  Expected Outcome:  Draw flowchart, write algorithm and write program for given scenario. Also attach the  screenshot of output.  **a.** )Write something about your characteristics not more than 50 words using gets function and print out the same using puts function.  Algorithm:  **Step 1 :** Start  **Step 2 :** Input  Enter text  **Step 3 :** Output : text  **Step 4 :** End  Flowchart:    Code:  #include<stdio.h> #include<conio.h> #include<string.h> void main() {   char line[500];   clrscr();    gets(line) ;   printf("your information is : \n"); puts(line);    getch();   }  Output Screen shot:    **Questions:**    1. What is the significance of using gets and puts? Are they acting as replacement of any function? How?  Answer: The gets() functions are used to read string input from the keyboard and puts() function displays it  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.  Hint: Use %o and %x  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 | Inputs | Octal | HexaDecimal | | 1 | Your Roll number | 117 | 4f | | 2 | 143 | 217 | 8f | | 3 | 0 | 0 | 0 | | 4 | 1 | 1 | 1 | | 5 | -1 | 17777 | ffff |   Algorithm:  Step 1: Start  Step 2: Get the value from user.  Step 3: Convert the value into octal using octal=%o, & hexadecimal using hexadecimal=%x & print the values.  Step 4: Stop  Flowchart:  D:\manav\ccp\4\4.1(b)flow.png  Code:  #include<stdio.h>  #include<conio.h>  void main()  {      int r;      printf("Enter your Number : ");      scanf("%d" ,&r);      printf("\n Hexadecimal of %d is %x\n",r,r);      printf("\n Octaldecimal of %d is %o",r,r);  }  Output Screenshot:  C:\Users\Admin\AppData\Local\Microsoft\Windows\INetCache\Content.Word\1.png |
| 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)  Expected Outcome:  Draw flowchart, write algorithm and write program for given scenario    Algorithm:  1. start the procedure  2. define c=7 and r=10  3. declare r,c,y  4. initialize r=1  5. print ------------------------------------------- and go next line  6. print multiplication table and go to next line  7. print ------------------------------------------- and go next line  8. enter in do while loop  9. initialize c=1  10. enter in do while loop  11. take input y=r\*c  12. print y  13. increment c by c++  14. check while condition  15. if it is true then go to step 10 otherwise directly go to step 16  16. then go to next line  17. increment r by r++  18. check while condition  19. if it is true then go to step 8 otherwise directly go to step 20  20. print -------------------------------------------  21. stop the procedure  Flowchart:    Code:  #include<stdio.h>  #include<conio.h>  #define C 7  #define R 10  void main()  {  int r,y,c;  r = 1;  clrscr();  printf("--------------------------------------------------");  printf("\n\tMULTIPLICATION TABLE\n");  printf("--------------------------------------------------\n");  printf("\n");  do  {  c= 1;  do  {  y = r\*c;  printf("%d\t",y);  c++;  }  while(c<=C);  printf("\n");  r++;  }  while(r<=R);  printf("--------------------------------------------------");  getch();  }  Output Screen shot: |
| **Set 5** | | | | |
| **Aim** | Write programs using If, If-else, If-else-if, Nested If, break, continue, goto and switch statements. | | | |
| **5.1** | Write a C program for the given scenario from the flowchart. Note that you have to enter your own height in centimeters. | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare height  Step 3 : Input height from keyboard  Step 4 : check if Height<150 Print you are dwarf and go to step 8  Step 5 : else Check Height>=150&&<=165 Print height is average and go to step8  Step 6 : else Check Height>=165 && <=195 Print you are tall and go to step 8  Step 7 : else Print height is abnormal  Step 8 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<conio.h>  void main()  {  int height,i;  for(i=1;i<=4;i++){  printf("Enter your height in cm : ");  scanf("%d",&height);  if(height<150)  {  printf("You are dwarf\n\n");  }  else if(height>=150 && height<165)  {  printf("Your height is average\n\n");  }  else if(height>=165 && height<195)  {  printf("You are tall\n\n");  }  else  {  printf("Your height is abnormal\n\n");  }  }  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | Tick marks your achieved result in the appropriate column:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Sr No. | Input (cm) | Dwart | Average | Tall | Abnormal | | 1. | Your height |  |  | yes |  | | 2. | Your mother’s height |  | yes |  |  | | 3. | Your father’s height |  |  | yes |  | | 4. | Your sibling’s | 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. Hint: Discriminant > 0 root1 = (-b + sqrt(discriminant)) / (2\*a) root2 = (-b - sqrt(discriminant)) / (2\*a) Discriminant < 0 root1 = root2 = -b / (2\*a) imaginary = sqrt (-discriminant) / (2\*a) (eg. Print it as: i20.3, i.e. i followed by value) Discriminant = 0 root1 = root2 = -b / (2\*a) | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare all variable  Step 3 : Take a, b, and c from user  Step 4 : D=b\*b-4\*a\*c  Step 5 :Check if D>0 go to step 6 else go to step 8  Step 6 : root1 =((-1)\*b)+sqrt(D))/2\*a, root2 =((-1)\*b)-sqrt(D))/2\*a  Step 7 : Print root 1 and root 2 and go to step 13  Step 8 :Check if D<0 go to step 9 else go to step 11  Step 9 : root1=root2=((-1)\*b)/2\*a; i= sqrt((-1)\*D)/2\*a  Step 10 : Print root=i\*root1 and go to step 13  Step 11 : root1 = root2 = (-b)/2\*a  Step 12 : print root  Step 13 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<math.h>  void main()  {  float a,b,c,D,j,root1,root2,i;  for (j=1;j<=3;j++)  {printf("Enter value of a:");  scanf("%f",&a);  printf("Enter value of b:");  scanf("%f",&b);  printf("Enter value of c:");  scanf("%f",&c);  D = (b\*b)-4\*a\*c;  switch(D>0)  {  case 1: root1 = ((-1)\*b + sqrt(D))/2\*a;  root2 = ((-1)\*b - sqrt(D))/2\*a;  printf("Root1 = %.2f\nRoot2 = %.2f\n\n",root1,root2);  break;  case 0:  switch(D<0)  {  case 1: root1=root2=((-1)\*b)/2\*a;  i= sqrt((-1)\*D)/2\*a;  printf("root = %.2fi%.2f (root is imaginary)\n\n",i,root1);  break;  case 0:  switch(0)  {  case 0: root1=root2=((-1)\*b)/2\*a;  printf("root = %.2f\n\n",root1);  }  }  }  }  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | **1. Have you learned about how to use normal switch case and nested switch case?**  Ans: Yes, we have learned the concept of using switch and nested switch case.  **2. Is default case necessary for every switch case?**  Ans: No default is not necessary for every switch case.  **3. What if break statement is not mentioned between two consecutive cases?**  Ans : If break is not defined the condition after satisfied condition will also be  executed. | | | |
| **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”** | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare variable Ram, Shyam , Ajay  Step 3 : Take input of ages from user  Step 4 : Check if Ram==Shyam==Ajay print all have same age and go to step 11  Step 5 : else Check if Ram==Shyam print Ram and Shyam have same ege and go  to step 11  Step 6 : else Check if Ajay==Shyam print Ajay and Shyam have same ege and go  to step 11  Step 7 : else Check if Ram==Ajay print Ram and Ajay have same ege and go  to step 11  Step 8 : else Check Ram<Shyam && Ram<Ajay print Ram is youngest and go to  step 11  Step 9 : else Check Shyam<Ram && Shyam<Ajay print Shyam is youngest and go  To step 11  Step 10 : else Check Ajay<Shyam && Ajay<Ram print Ajay is youngest and go to  step 11  Step 11 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<conio.h>  void main()  {  int Ram,Shyam,Ajay;  printf("Enter the ages of Ram:");  scanf("%d",&Ram);  printf("Enter the ages of Shyam:");  scanf("%d",&Shyam);  printf("Enter the ages of Ajay:");  scanf("%d",&Ajay);  if(Ram==Shyam && Shyam==Ajay)  {  printf("All of them have same age.\n");  }  else if(Ram==Shyam)  {  printf("Ram and Shyam have same age.\n");  }  else if(Shyam==Ajay)  {  printf("Shyam and Ajay have same age.\n");  }  else if(Ram==Ajay)  {  printf("Ram and Ajay have same age.\n");  }  else  {  if(Ram<Shyam && Ram<Shyam)  {  printf("Ram is the youngest of all.\n");  }  else if(Shyam<Ram && Shyam<Ajay)  {  printf("Shyam is the youngest of all.\n");  }  else if(Ajay<Ram && Ajay<Shyam)  {  printf("Ajay is the youngest of all.\n");  }  }  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | **1. Have you tried merging the concepts of Nested if else and else if ladder in this**  **scenario?**  Ans : Yes, we have used the concept of merging of Nested if else and else if  ladder in this scenario.  **2. Differentiate the concept of Nested if else and else if ladder.**  Ans : The major difference between the nested if else and else if ladder is that in  nested one if stamen is written in other and it goes inside and inside while I  ladder it is written in ladder structure that is one after the another | | | |
| **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.** | | | |
| **Algorithm** | Step 1 : Start  Step 2 : input order item and credit  Step 3 :check If(order<=100 && credit==’n’|| credit==’N’) go to step 4 else  go to step 5  Step 4 : print We will deliver you the product and go to step 8  Step 5 : Chevk if (Order>100) go to step 6 else go to step 7  Step 6 : print Order is out of stock and go to step 8  Step 7 : print Please complete your credit first  Step 8 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<conio.h>  void main()  {  int order=100;  char credit;  printf("Enter the number of items you want to order:");  scanf("%d",&order);  printf("Enter your credit(Y or N):\n");  credit=getch();  if(order<=100 && credit=='n' || credit=='N')  {  printf("We will deliver you the product.");  }  else if(order>100)  {  printf("Order out of stock.");  }  else  {  printf("Please complete your credit first.");  }  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | **Which kind of logic have you used for building this program? If else if ladder**  **or nested if else statements?**  Ans. We use else if ladder logic for building this program. | | | |
| **Set 6** | | | | |
| **Aim** | **Write programs using While loop, Do-while, simple for loop, nested for loop, break and continue.** | | | |
| **6.1** | **There is a person, who is asked to enter the alphanumeric password for registeringinto 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.** | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare string and variable u=0, l=0, n=0, i=0  Step 3 : Take password from user  Step 4 : Check if i<=strlen(pass) go to step 5 else go to step 9  Step 5 : Check if isupper(pass[i]) do u=1 else go to step 6  Step 6 : Check if islower(pass[i]) do l=1 else go to step 7  Step 7 : Check if isdigit(pass[i]) do n=1 else go to step 8  Step 8 : do i++ and go to step 4  Step 9 : Check if (u=0||l=0||n=0) print Password does not satisfy constraints!! Please try  again and go to step 3  Step 10 : else print Good password, you may proceed  Step 11 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<string.h>  #include<ctype.h>  void main()  {  char pass[20];  int i,u,l,n;  do{  u=0,l=0,n=0;  printf("Enter a pass word\n");  scanf("%s",&pass);  for(i=0;i<strlen(pass);i++)  {  if(isupper(pass[i]))  {  u=1;  }  if(islower(pass[i]))  {  l=1;  }  if(isdigit(pass[i]))  {  n=1;  }  }  if( u==0 || l==0 || n==0 )  {printf("Password does not satisfy constraints!! Please try again\n\n");  }  else  {  printf("Good Password,you may proceed");  }  }while(u==0 || l==0 || n==0);  printf("\n\nThis program is prepared by yagnik ladani\nID=21CE062\n");  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | **Have you understood working of do...while loop? Do mention the syntax of**  **this loop.**  Ans : Yes, we understood the working of the do…while loop. First the statement  written in the do will first executed then, before entering the while loop the  condition is checked and then it will get executed.  **2. Have you used for loop in this program?**  Ans : Yes, we have used the for loop in the above program.  3**. What is goto statement? How is it useful?**  Ans : Goto statement is used to directly move the compiler point to the point  where the function is defined. | | | |
| **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.** | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare variable bas, pow, ans=1, i=0  Step 3 : Input bas and pow from key borad  Step 4 : Check if i<pow go to step 5 else go to step 6  Step 5 : do ans = ans \* bas and i++ and go to step 4  Step 6 : Print ans  Step 7 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<conio.h>  void main()  {  int bas,power,ans=1,i=1;  printf("Enter the value of base\n");  scanf("%d",&bas);  printf("Enter the value of power\n");  scanf("%d",&power);  while(i<=power)  {  ans=ans\*bas;  i++;  }  printf("Sr no. base no. power output\n");  printf("1 %d %d %d\n",bas,power,ans);  printf("2 3 3 27\n");  printf("3 7 2 49\n");  printf("\n\nThis program is prepared by yagnik ladani\nID=21CE062\n");  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | **1. Have you understood the concept of while loop? if yes write its syntax here.**  Ans : Yes, I understood the concept of while loop, I the while loop the loop will  executed until the condition is satisfied.  While(condition)  {  Statement  } | | | |
| **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** | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare all variable i=1,amo,tot\_amo=0  Step 3 : check if i>0 go to step 4  Step 4 : input amount from key board  Step 5 : tot\_amo=tot\_amo+amo and do i++  Step 6 : check If amo == 0 go to step 7 else go to step 3  Step 7 : Print tot\_amo  Step 8 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<conio.h>  void main()  {  int i,amo,tot\_amo=0;  printf("Sr no.\tEnter amount\n");  for(i=1;i>0;i++)  {  printf("%d \t",i);  scanf("%d",&amo);  tot\_amo=tot\_amo+amo;  if(amo==0)  break;  }  printf("Total amount = %d",tot\_amo);  printf("\n\nThis program is prepared by yagnik ladani\nID=21CE062\n");  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | 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.**  Ans : Yes, we have learnt the concept of looping for repeating the certain code.  Here , the i stands for number of time of the loop is executed . | | | |
| **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.** | | | |
| **Algorithm** | Step 1 : Start  Step 2 : Declare variable sum=21,i=1  Step 3 : Check if i>0 Input matchstick  Step 4 : Check if matchstick>4 print value is invalid and go to step 3  Step 5 : else do c=5-a and print it  Step 6 : Do sum=sum-a-c and print it  Step 7 : Check if sum ==1 Print Now, You have to picked up last 1 sticks, so you are  a looser… and go to step 8 else go to step 3  Step 8 : Stop | | | |
| **Flowchart** |  | | | |
| **Code** | #include<stdio.h>  #include<conio.h>  void main()  {  int a,b,i,sum=21,tot,c;  for(i=1;i>=0;i++)  {  printf("Enter number of match-sticks =");  scanf("%d",&a);  if(a>4)  {  printf("value is invalid\n\n");  continue;  }  c=5-a;  printf("Value enter by computer = %d\n",c);  sum=sum-a-c;  printf("Reamning match-sticks= %d\n\n",sum);  if(sum==1)  break;  }  printf("Now, You have to picked up last 1 sticks, so you are a looser...");  getch();  } | | | |
| **Output** |  | | | |
| **Question answers** | 1**. What is the significance of using break and continue statement?**  Ans : Break statement is used to come out of loop entirely while continue  statement is used to skip the particular statement. | | |
| **Set 7** | | | |
| **Aim** | **Write programs on arrays (Sorting, Merging,finding particular value, etc.)** | | |
| **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?** | | |
| **Algorithm** |  | | |
| **Flowchart** |  | | |
| **Code** | #include<stdio.> #include<conio  int main()  {  int a[90],i,n,pos=0,neg=0,even=0,odd=0; printf("enter the number of n:"); scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d",&a[i]);  }  for(i=0;i<n;i++)  {  if(a[i]>0)  {  pos++;  }  if(a[i]%2==0)  {  even++;  }  if(a[i]<0)  {  neg++;  }  if(a[i]%2!=0)  {  odd++;  }  }  printf("positive=%d\nnegative=%d\nodd=%d\neven=%d",pos,neg,odd,even); return 0;  } | | |
| **Output** |  | | |
| **Question answers** | 1**. 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 )Because C++ doesn't automatically set it zero for you. So, you should initialize it Yourself, int c = 0saw the Author of the course/tutorial zero-initialize a variable and started wondering. why is zero-initializing even an option? You could just identify the variable and assign a value later. There a performance benefit | | |
| **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.** | | |
| **Algorithm** |  | | |
| **Flowchart** |  | | |
| **Code** | #include<stdio.h> #include<conio.h> int main()  {  int a[90],i,n,A,B,c[90],b[90],k,temp,j; printf("enter the number of A:"); scanf("%d",&A);  for(i=0;i<A;i++)  {  scanf("%d",&a[i]); c[i]=a[i];  }  k=i;  printf("enter the number of B:"); scanf("%d",&B); for(i=0;i<B;i++)  {  scanf("%d",&b[i]); c[k]=b[i];  k++;  }  for(i=0;i<k;i++)  {  for(j=i+1;j<k;j++)  {  if(c[j]<c[i])  {  temp=c[i]; c[i]=c[j]; c[j]=temp;  }  else  {  continue;  }  }  }  printf("merge and short array of A and B\n"); for(i=0;i<k;i++)  {  printf("%d ",c[i]);  }  return 0;  } | | |
| **Output** | https://lh6.googleusercontent.com/J9IHjNALqkKEgA-radDp0f04hsewL93qp3PiVK2xZmY5H-FwlsHKsDaK0XrfESyMvtQrxn22XbHTzfdYaWb8j7_qrE4MJG7MROB4e3OtdUXjtK1nqIcZ3EUfG5wM_oIw_c1CKfej | | |
| **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** | | |
| **Algorithm** |  | | |
| **Flowchart** |  | | |
| **Code** | #include<stdio.h> void main()  { int a[3][3],i,j,b[3][3];  printf("enter the value for matrix A:\n"); for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  scanf("%d",&a[i][j]);  }  }  printf("enter the value for matrix b:\n"); for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  scanf("%d",&b[i][j]);  }  }  for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  for(k=0;k<3;k++)  {  c[i][j]+=(a[i][k]\*b[k][i]);  }  }  }  printf("A\*B=\n"); for(i=0;i<3;i++)  {  for(j=0;j<3;j++)  {  printf("%5d",c[i][j]);  }  printf("\n");  }  } | | |
| **Output** | https://lh4.googleusercontent.com/mE5SowsXlUd5wy4qVCeecBEDWIbSYT4gMbL9xFwUveZMaJhmAwuadjQ6VAq7x-W20WgGkgKD_DpOoKPXIAy6aCevvnry5dNiEkptGDFyWdHySOkb7JaxTUj2_BsB_FTr_7vXTgH3 | | |
| **Question answers** | **State the advantages of using Array Indexes. When is it suitable to take array index?**  Ans:  In an array, accessing an element is very easy by using the index number. The search process can be applied to an array easily. 2D Array is used to represent matrices. For any reason a user wishes to store multiple values of similar type then the Array can be used and utilized efficiently | | |
|  |  | | |
| **Set-8** | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 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). | | | | | | | |
| **Sr. No.** | **Input** | **Lowercase** | **Uppercase** | **Digits** | **Symbol** | **Output** |  |
| Example | Abc@1234 | ✓ | ✓ | ✓ | ✓ | Strong |  |
| 1. |  |  |  |  |  |  |  |
| 2. |  |  |  |  |  |  |  |
| 3. |  |  |  |  |  |  |  |
| **Algorithm:**  **Flow chart:**  **Code:**  #include<stdio.h>  #include<conio.h>  void main()  {  int i,lower,upper,num,spchar;  i = lower = upper = num = spchar = 0;  char str[10];  printf("enter your password : ");  scanf("%s",str);  while(str[i]!='\0')  {  if(str[i]>='a'&&str[i]<='z')  {  lower++;  }  if(str[i]>='A'&&str[i]<='Z')  {  upper++;  }  if(str[i]>=48&&str[i]<=57)  {  num++;  }  if(str[i]>=33&&str[i]<=47)  {  spchar++;  }  if(str[i]>=58&&str[i]<=64)  {  spchar++;  }  i++;  }  if(lower>=1&&upper>=1&&spchar>=1&&num>=1&&spchar>=1)  {  printf("Strong password");  }  if(lower>=1&&upper<1&&spchar>=1&&num>=1&&spchar>=1)  {  printf("average password");  }  if(lower<1&&upper<1&&spchar>=1&&num>=1&&spchar>=1)  {  printf("average password");  }  }  Output screenshot:    **Questions:**   1. Explain the difference between string and character. Also write the syntax for printing character and string.   String example = "Here you can have anything"; Well, char (or its wrapper class Character) means a single character, i.e. you can't write 'ab' whereas String is a text consisting of a number of characters and you can think of a string a an array of characters (in fact the String class has a member char[] value). | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 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. | | | | |
| **Sr. No.** | **Input of names** | **Sorted Output as per output** | |  |
| 1. |  |  | |
| 2. |  |  | |
| 3. |  |  | |
| 4. |  |  | |
| 5. |  |  | |
| **Algorithm:**  Step 1: First, create an array and initialize with the values.  Step 2: For loop from i=0 to i<size\_of\_array:  Step 3: Nest another for loop from j=0 to j<size\_of\_array-i-1:  Step 4: Check if(strcmp(array[j], array[j+1]) > 0):  Step 5: If yes, then swap(array[j], array[j+1])  Step 6: End nested loop.  Step 7: End external loop.  Step 8: Output sorted array  **Flowchart:**  **Code:**  **#include<stdio.h>**  **#include<string.h>**  **main(){**  **int i,j,n;**  **char str[100][100],s[100];**  **printf("Enter number of names :\n");**  **scanf("%d",&n);**  **printf("Enter names in any order:\n");**  **for(i=0;i<n;i++){**  **scanf("%s",str[i]);**  **}**  **for(i=0;i<n;i++){**  **for(j=i+1;j<n;j++){**  **if(strcmp(str[i],str[j])>0){**  **strcpy(s,str[i]);**  **strcpy(str[i],str[j]);**  **strcpy(str[j],s);**  **}**  **}**  **}**  **Output Screen Shot:**  **printf("\nThe sorted order of names are:\n");**  **for(i=0;i<n;i++){**  **printf("%s\n",str[i]);**  **}**  **}**    **Questions:**  1. Which string functions have you learned from this program? Explain any 5 string functions in below given table. | | | | |
| **Sr. No.** | **String Functions Syntax** | | **Purpose** |  |
| 1. | strcpy | | To copy string |
| 2. |  | |  |
| 3. |  | |  |
| 4. |  | |  |
| 5. |  | |  |
|  | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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. | | | | | | | | | | | | | |
| **Sr. No.** | | | **Input** | | | **Sorted Output as per output** | | | | |  | | |
| 1. | | | Alpha | | |  | | | | |
| 2. | | | Madam | | |  | | | | |
| 3. | | | saippuakivikauppias | | |  | | | | |
| 4. | | | [Hannah](https://www.dictionary.com/browse/Hannah) | | |  | | | | |
| **Algorith:**  Step 1. Start  Step 2. Read the string from the user  Step 3. Calculate the length of the string  Step 4. Initialize rev = “ ” [empty string]  Step 5. Initialize i = length - 1  Step 6. Repeat until i>=0:  6.1: rev = rev + Character at position ‘i’ of the string  6.2: i = i – 1  Step 7. If string = rev:  7.1: Print “Given string is palindrome”  Step 8. Else:  8.1: Print “Given string is not palindrome”  Step 9. Stop  **Flowchart:**    **Code:**  #include <stdio.h>  #include <string.h>    int main()  {  char str[100];  int i, len, flag;  flag = 0;    printf("\n Please Enter any String : ");  gets(str);    len = strlen(str);    for(i = 0; i < len; i++)  {  if(str[i] != str[len - i - 1])  {  flag = 1;  break;  }  }  if(flag == 0)  {  printf("\n %s is a Palindrome String", str);  }  else  {  printf("\n %s is Not a Palindrome String", str);  }    return 0;  }  Output screenshot:    **Questions:**   1. Explain the concept of recursion. Explain the difference between recursion and iteration?   Overhead: **Recursion has a large amount of Overhead** as compared to Iteration. Recursion: Recursion has the overhead of repeated function calls, that is due to repetitive calling of the same function, the time complexity of the code increases manifold. Iteration: Iteration does not involve any such overhead. | | | | | | | | | | | | | |
| 9 | | 9.1 | | Write a C program to check if the entered number is prime or not by using types of user defined functions   1. No arguments passed and no return value 2. No arguments passed but a return value 3. Argument passed but no return value 4. 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: | | | | | | | | | | | |
| **Sr. No.** | **User Defined Functions** | | | **Input** | | **Prime** | | **Non-Prime** |  | | |
| 1. | No arguments passed and no return value | | |  | |  | |  |
| 2. | No arguments passed but a return value | | |  | |  | |  |
| 3. | Argument passed but no  return value | | |  | |  | |  |
| 4. | Argument passed and a return value | | |  | |  | |  |
| **Algorithm:**  **FlowChart:**  **Code:**  #include<stdio.h>  #include<conio.h>  void prime();  void main()  {  clrscr();  printf("enter the number:");  prime();  getch();  }  void prime()  {  int i,num,j;  scanf("%d",&num);  for(i=2;i<num;i++)  {  if(num%i==0)  {  j=1;  break;  }  }  if(j==1)  printf("number is not prime.");  else  printf("number is prime.");  }  **Output:**    **Questions:**  1. You might be clear now, how user defined functions are created in different ways. Explain them. | | | | | | | | | | | |
|  | | 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.  Algorithm:  Flowchart:  Code:  #include<stdio.h>  #include<conio.h>  #include<math.h>  void check(int a,int b,int c);  void area(float, float, float);  void main()  {  int a,b,c;  printf("Enter the value of side length a:");  scanf("%d",&a);  printf("\nEnter the value of side length b:");  scanf("%d",&b);  printf("\nEnter the value of side length c:");  scanf("%d",&c);  check(a,b,c);  getch();  }  void check(int a,int b, int c)  {  if(a+b>c || b+c>a || a+c>b)  {  printf("It will form a triangle.\n");  area(a,b,c);  }  else  {  printf("It will not form triangle.\n");  }  }  void area(float a, float b, float c)  {  float s,A;  s=(a+b+c)/2;  A=sqrt(s\*(s-a)\*(s-b)\*(s-c));  printf("Area of triangle is :%f",A);  }  Output: | | | | | | | | | | | |
| **Sr. No.** | | | **Input** | **Forming Triangle** | | | **Not a Triangle** | | | |  |
|  | | |  |  | | |  | | | |  |

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Input** | **Binary** |
| 1. |  |  |
| 2. |  |  |
| 3. |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  | |  |  |  | |  | |  |
|  | | **a** | | **b** | **c** |  | |  | |  |
| 1. | |  | |  |  |  | |  | |  |
| 2. | |  | |  |  |  | |  | |  |
| 3. | |  | |  |  |  | |  | |  |
| **Questions:**  1. Explain the concept of nested functions in C. | | | | | | | | | | |
|  | 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:  Algorithm:  Flowchart:  Code:  #include<stdio.h>  int binary(int n)  {  if(n==0)  return 0;  else  return((n%2)+10\*binary(n/2));  }  int main()  {  int a;  printf("Enter positive number :\n");  scanf("%d",&a);  printf("Binary Number is=%d",binary(a));  }  Output:    **Questions:**  1. Mention the advantages of using recursion in a program. | | | | | | | | | | |
| 10 | 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: | | | | | | | | | | |
|  | **Sr. No.** | | **Book Title** | | | | **Author Name** | | **Amount of book** | |
| 1. | |  | | | |  | |  | |
| 2. | |  | | | |  | |  | |
| 3. | |  | | | |  | |  | |
| **Algorithm:**  **Flowchart:**  **Code:**  #include<stdio.h>  struct book  {  char Book\_title[20];  char Author\_name[20];  float Amount;  };  void bookDetails(struct book b)  {  printf("Book name : %s\n",b.Book\_title);  printf("Author name :%s\n",b.Author\_name);  printf("Amount of Book :%f\n",b.Amount);  }  int main()  {  struct book b;  printf("Enter book name :");  scanf("%s",b.Book\_title);  printf("Enter name of author :");  scanf("%s",b.Author\_name);  printf("Enter amount :");  scanf("%f",&b.Amount);  printf("\n");  bookDetails(b);  return 0;  }  **Output:**    **Questions:**  1. Can we declare function inside structure of C Programming? Explain Why? | | | | | | | | | | |
|  | 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:** | | | | | | | | | | |

|  |  |
| --- | --- |
| Marks | Categories |
| 70 or above | DISTINCTION |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Accession Number** | **Title of Book** | **Author** | **Price** | **Flag** | **Output** |
| 1. |  |  |  |  |  | **Book Issued** |
| 2. |  |  |  |  |  | **Book Not Issued** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  | 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.  **Algorithm:**  **Flowchart:**  **Code:**  #include<stdio.h>  union librery  {  char Book\_title[20];  char Auther\_name[20];  int price;  int Accession\_num;  int flag;  };  int main()  {  union librery b;  printf("Enter Book Name :");  gets(b.Book\_title);  printf("Enter Author Name :");  scanf("%s",b.Auther\_name);  printf("Enter price :");  scanf("%d",&b.price);  printf("Enter accession number :");  scanf("%d",&b.Accession\_num);  printf("Enter 1 for book issued else 0 :");  scanf("%d",&b.flag);  if(b.flag)  {  printf("Book issued");  }  else  {  printf("Not issued");  }  return 0;  **}**  **Output:**    **Questions:**  1. Explain the major difference between structure and union in detail. |
|  | 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.  Get the output as mentioned below:    **Algorithm:**  **Flowchart:**  **Code:**  **Output:**  **Questions:**  1. Explain how nested structure works in C programming. |
| 11 | 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. |

|  |  |
| --- | --- |
| 69 to 60 | FIRST CLASS |
| 59 to 40 | PASS |
| Below 40 | FAIL |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | For example, if following marks of 10 students are entered:  34 56 78 98 12 31 67 75 91 23  Then the output should be  DISTINCTION 4 FIRST CLASS 1 PASS 1 FAIL 4  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  You are requested to gain all categories of results, so input the values accordingly, also write the counts of all the categories. | | | | | | |
|  | **Sr. No.** | **Input** | **Distinction** | **First Class** | **Pass** | **Fail** |
| 1. |  |  |  |  |  |
| 2. |  |  |  |  |  |
| ... |  |  |  |  |  |
| 10. |  |  |  |  |  |
|  | **Counts** |  |  |  |  |
| **Algorithm:**  **Flowchart:**  **Output:**  **Questions:**  1. Explain the importance of using pointers? | | | | | | |
|  | 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!!"); | | | | |  |
|  |  | } | | | | |  |

|  |  |  |
| --- | --- | --- |
|  |  | **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);  } |
| 12 | 12.1 | Write a program to read a text file ‘Demo.txt’ and print each word of that file in reverse order.  **Expected Output:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Example:  Input: HELLO Output: OLLEH  Algorithm:  Flowchart:  Code:  Output:  **Questions:**  1. Explain, why do we need to use files in C? |
|  | 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.  **Expected Outcome:** |

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Function** | **Purpose** |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the data in a file from console and retrieve that data on the console. Also attach the screenshot of file where the data is written.  **Questions:**  1. Explain any 3 functions of file other then mentioned in the problem. | | | | | | |
|  | 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.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the data in a file from console and attach the screenshots of Data1.txt, Data2.txt and Data3.txt files. Also add the screenshot of console.  **Algorithm:**  **Flowchart:**  **Code:**  **Output:**  **Questions:**  1. Explain the difference between argc and argv along with their significance. | | | | | | |
| 13 | 13.1 | Write a program to read and print the student details using structure and Dynamic Memory Allocation.  Following student details needs to be included:  Roll No., Name, Age, Class, Branch.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter this student details for N number of students, collect the no. of details to be entered from the user and ask for that many student’s details. Enter all details in below mentioned table and print the values collected from user. | | | | | | |
|  | **Sr. No.** | **Roll No.** | **Name** | **Age** | **Class** | **Branch** |
| 1. |  |  |  |  |  |
| 2. |  |  |  |  |  |
| ... |  |  |  |  |  |
| N. |  |  |  |  |  |
| **Algorith:**  **Flowchart:**  **Code:**  **Output:**  **Questions:**  1. Explain the benefits of using dynamic memory allocation. Give one scenario where it is most useful. | | | | | | |
|  | 13.2 | Write a program using a character string in a block of memory space created by **calloc** | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **()** and then modify the same to store a larger string using **realloc ()** function. **(Dynamic Array)**.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the details in below given table as per the requirement: | | | | |
|  | **Sr. No.** | **Instruction** | | **Output** |
| 1. | **String to be entered** | |  |
| 2. | **String received after reallocation of memory** | |  |
| **Questions:**  1. Mention advantage of using realloc() function. | | | | |
|  | 13.3 | Write a program to enter N numbers into array and find average. **Enter the size of the array through keyboard. (Dynamic Array)**. Use malloc () to allocate memory and use **free()** to free the memory after the use.  **Expected Outcome:**  Draw flowchart, write algorithm and program for given scenario. Also attach screenshot of output.  Enter the details in below given table as per the requirement: | | | | |
|  | **Sr. No.** | **Instruction** | **Output** | |
| **Enter the size of Array** | | **N (To be entered by user)** | |
| 1. |  | **To be entered by user** | |
| 2. |  | **To be entered by user** | |
| … |  | **To be entered by user** | |
| N. |  | **To be entered by user** | |
| **Average of entered values** | |  | |
|  | |  | |