

PROGRAMMING IN C LAB MANUAL

Subject Code:

Class: I Year I Semester

**Prepared By**

Dr. M.N. Nachappa, Professor,

School of CS and IT,

Jain (Deemed-to-be) University

PROGRAMMING IN C LABORATORY OBJECTIVE

The course is designed to provide a practical exposure to the students on C language. It helps students understand the concept of a C program like variables, control structures, arrays, functions, pointers, macro processor, files.

OUTCOMES

* Upon completion of the course, the students acquire the knowledge to build the logic and develop a solution for a problem statement in C-language.
* Understand the use of structured program development in C as applied to both large software systems and to small programming projects.
* Understand the use of arrays, functions, pointers, macro processors, structures, unions, files
* Understand the use and structure, pointers and files

Program List

|  |  |  |
| --- | --- | --- |
| Session No. | Programs | Page No. |
| 1 | 1. C program to display Your Name, Address and City in different lines. 2. C program to find the area and circumference of a circle. 3. C program to convert centigrade into Fahrenheit. Formula: **F= (1.8 \* C) +32.** 4. C program to swap variable values of two variables    1. Using a temporary variable    2. Without Using a temporary variable 5. C program to calculate the total salary of an employee given Allowance1 is 33% of Basic Pay, Allowance2 is 73% of Basic Pay and Deduction is 52% of Basic Pay. 6. C program to calculate simple interest. |  |
| 2 | 1. C program to find the largest of three numbers (if). 2. C program to check whether a given year is a leap year (if-else). 3. C program to find the largest, smallest and second largest of three numbers. 4. C program to find the second largest and second smallest of four numbers (else-if). 5. C program to output the next date for a given date (else-if). 6. C program to find the roots of a quadratic equation (else-if) 7. C program to check whether a given date is valid or   not (switch). |  |
| 3 | 1. C program to output the digits of a number (while). 2. C program to find the sum of all numbers from 1 to   “N” (while).   1. C program to reverse a number (while). 2. C program to calculate compound interest (do-while). 3. C program to convert from (do-while)    1. Decimal to binary    2. Binary to decimal |  |
| 4 | 1. C program to find the factorial of a number (for). 2. C program to check whether a number is prime or not (for). 3. C program to generate first “N” Fibonacci numbers   (for). |  |

|  |  |  |
| --- | --- | --- |
|  | 1. C program to calculate x^y (for). 2. C program to find the sum of the series (for)    1. SIN(X)    2. COS(X) |  |
| 5 | 1. C program to generate prime numbers from 1 to “N”. 2. C program to generate the patterns. a)1 b) 1   1 2 2 2  1 2 3 3 3 3  1 2 3 4 4 4 4 4   1. C program to generate the patterns.   a)4 3 2 1 b) 4 4 4 4  3 2 1 3 3 3  2 1 2 2  1 1   1. C program to generate the patterns. a)1 b) 1   1 2 2 2  1 2 3 3 3 3  1 2 3 4 4 4 4 4  1 2 3 3 3 3  1 2 2 2 2  1 1   1. C program to generate the pattern a)1 2 3 4 b)1   2 3 4 5 2 3  3 4 5 6 4 5 6  4 5 6 7 7 8 9 10 |  |
| 6 | 1. C program using functions to find GCD and LCM of two numbers. 2. C program using functions to convert a decimal number to its binary equivalent. 3. C program using functions to convert a binary number to its decimal equivalent. 4. C program using functions to check whether a three digit number is an Armstrong number or not. 5. C program using functions to calculate compound   interest. |  |
| 7 | 1. C program using recursion to find the factorial of a number. 2. C program using recursion to find x^y. 3. C program using recursion to find the Nth Fibonacci |  |

|  |  |  |
| --- | --- | --- |
|  | number.   1. C program using recursion to find the sum of natural numbers. 2. C program using recursion to count the digits of a   number. |  |
| 8 | 1. C program to find the largest and smallest of “N”   numbers using 1-D arrays.   1. C program to sort a list of numbers using Bubble sort. 2. C program to sort a list of numbers using Selection sort. 3. C program to sort a list of numbers using Insertion sort. 4. C program to find search for a given number using Linear search. 5. C program to find search for a given number using   Binary search. |  |
| 9 | 1. C program to find the sum of two matrices. 2. C program to find the product of two matrices. 3. C program to transpose a given matrix. 4. C program to check whether a given matrix is an identity matrix. 5. C program to check whether a given matrix is a scalar   matrix. |  |
| 10 | 1. C program to find the frequency of a character in a string. 2. C program to reverse a string. 3. C program to copy the contents of one string to another. 4. C program to check whether a given string is a palindrome or not (without library functions). 5. C program to remove all blank spaces and punctuation   symbols from a string. |  |
| 11 | 1. C program to create and use a pointer. 2. C program to swap the values of two variables using pointers. 3. C program to find the area and circumference of a circle using pointers and functions. 4. C program to sort a list of numbers using pointers. 5. C program to concatenate two strings using pointers and functions. |  |
| 12 | 1. C program to create and use a structure for a student data. 2. C program to add two time periods using structures. |  |

|  |  |  |
| --- | --- | --- |
|  | 1. C program to create and use a union. 2. C program to create a file to hold the data of employees input and output data from it. 3. C program to write a sentence in a file and convert all   lower case alphabets to uppercase and vice versa. |  |

1. To display Your Name, Address and City in different lines.

|  |  |
| --- | --- |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main( )  {  clrscr( ); **// clears the output window**  printf("\n Name : Jain University"); printf("\n Address : Jayanagar”); printf("\n City : Bengaluru"); getch();  } |  |

Output:

2. To find the area and circumference of a circle.

|  |  |
| --- | --- |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main()  {  int rad;  float PI = 3.14, area, ci;  clrscr();  printf("\nEnter radius of circle: "); scanf("%d", &rad);  area = PI \* rad \* rad; ci = 2 \* PI \* rad;  printf("\nArea of circle : %f ", area); printf("\nCircumference : %f ", ci);  getch();  } |  |

Output:

3. To convert celsius into Fahrenheit. Formula: F= (1.8 \* C) +32.

|  |  |
| --- | --- |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main()  {  float celsius, fahrenheit;  clrscr();  printf("\nEnter temp in Celsius : "); scanf("%f", &celsius);  fahrenheit = (1.8 \* celsius) + 32; printf("\nTemp. in Celsius : %f ",  celsius); printf("\nTemp. in Fahrenheit : %f ",  fahrenheit);  getch();  } |  |

Output:

7. To find the largest of three numbers (if).

|  |  |
| --- | --- |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main()  {  int a, b, c,L ;  clrscr();  printf("\nEnter value of a, b & c : "); scanf("%d%d%d", &a, &b, &c);  if ((a > b) && (a > c)) L = a;  if ((b > c) && (b > a)) L = b;  if ((c > a) && (c > b)) L = c;  printf("\n%d is the largest number", L); getch();  } |  |

Output:

|  |  |
| --- | --- |
| **8. To check whether a given year is a leap year (if-else).** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main()  {  int y,r ;  clrscr();  printf("\nEnter any year : ");  scanf("%d”, &y);  r = y % 4; if (r==0)  printf(“ \n %d is a leap year “, y);  else  printf(“ \n %d is not a leap year “, y);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **9. To find the largest, smallest and second largest of three numbers.** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main()  {  int a, b, c;  int L,S,SL;  clrscr();  printf("\nEnter value of a, b & c : "); scanf("%d%d%d", &a, &b, &c);  L =a;  if (b > L) L = b;  if (c > L) L = c;  S =a;  if (b < S) S = b;  if (c < S) S = c; SL = (a+b+c) – (L+S);  printf("\n%d is the largest number", L); printf("\n%d is the smallest number", S); printf("\n%d is the second largest  number", SL);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **12. To find the roots of a quadratic equation (else-if)** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<math.h> #include<conio.h>  void main()  {  float a, b, c;  float desc, root1, root2;  clrscr();  printf("\nEnter the constants: "); scanf("%f%f%f", &a,&b,&c);  desc = b \* b - 4 \* a \* c; if ( desc > 0)  {  printf(“\n Roots are Real”)  root1 = (-b + sqrt(desc))/(2.0 \* a); root2 = (-b – sqrt(desc))/(2.0 \* a); printf("\nFirst Root : %f", root1); printf("\nSecond Root : %f",  root2);  }  else if ( desc == 0)  {  printf(“\n Roots are Equal”)  root1 = -b / (2.0 \* a); printf("\nThe Root is : %f",  root1);  }  else  {  printf(“\n Roots are Imaginary”)  root1 = -b / (2.0 \* a);  root2 = sqrt(abs(desc)) / (2.0 \* a); printf("\nReal part : %f", root1); printf("\nImaginary part : %f",  root2);  }  getch ();  } |  |

Output:

|  |  |
| --- | --- |
| **15. To find the sum of all numbers from 1 to “N” (while).** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> |  |

|  |  |
| --- | --- |
| #include<conio.h>  void main()  {  int i, num, sum;  clrscr();  printf("\n Enter the limit "); scanf ("%d", &num);  sum =0;  i = 1;  while (i <= num)  {  sum = sum + i;  ++i;  }  printf ("Sum of first %d natural numbers  = %d\n", num, sum);  getch();  } |  |

Output:

|  |  |  |
| --- | --- | --- |
|  | **16. To reverse a number (while).** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> |  |

|  |  |
| --- | --- |
| #include<conio.h>  void main()  {  int num, num1,rem, rev;  clrscr();  printf("\nEnter any number: "); scanf("%d", &num);  num1 = num; rev = 0;  while (num !=0)  {  rem = num % 10; rev = rev \* 10 + rem; num = num / 10;  }  printf("\n Number : %d", num1); printf("\nReversed Number : %d", rev);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **18.a. To convert from Decimal to binary (do-while)** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <math.h> #include<conio.h>  void main()  {  int num,dec,bin,r,k;  clrscr();  printf("Enter a Decimal number: "); scanf("%f", &dec);  num = dec; bin = 0;  k = 1;  do  {  r = num % 2; num = num /2; bin = bin + r \* k; k = k \* 10;  } while (num!=0);  printf("\n Decimal Number : %d", dec); printf("\nBinary Number : %d", bin);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **19. To find the factorial of a number.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> |  |

|  |  |
| --- | --- |
| #include <conio.h> void main()  {  int i, n, fact = 1;  clrscr();  printf("\nEnter a number"); scanf("%d", &n);  for (i = 1; i <= n; i++)  fact = fact \* i;  printf("Factorial of %d = %d\n", n, fact); getch();  } |  |

Output:

|  |  |
| --- | --- |
| **20. To check whether a given number is prime or not.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> |  |

|  |  |
| --- | --- |
| #include <conio.h> void main()  {  int n, i, flag = 1; clrscr();  printf("Enter a positive integer: "); scanf("%d", &n);  for(i = 2; i <= n/2; ++i)  {  if(n%i == 0)  {  flag = 0; break;  }  }  if (n == 1)  printf("1 is neither a prime nor a  composite number.");  else  {  if (flag)  printf("%d is a prime  number.", n);  else  printf("%d is not a prime  number.", n);  }  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **23.a. To find the sum of the series to calculate SIN(x).** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h>  #include<conio.h> |  |

|  |  |
| --- | --- |
| void main()  {  int i;  float x, sum, t;  clrscr();  printf(" Enter the value for x : "); scanf("%f",&x);  x=x\*3.14/180; t=x;  sum=x;  for(i=1;i<=n;i++)  {  t=(t\*(-1)\*x\*x)/(2\*i\*(2\*i+1)); sum=sum+t;  }  printf(" The value of Sin(%f) =  %.4f",x,sum);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **25.a. To generate the pattern 1**  **1 2**  **1 2 3** | |
| **Program** | **Algorithm and Flowchart** |

|  |  |
| --- | --- |
| #include <stdio.h> #include <conio.h>  void main()  {  int i, j, rows;  clrscr();  printf("Enter number of rows: "); scanf("%d",&rows);  for(i=1; i<=rows; i++)  {  for(j=1; j<=i; j++)  {  printf("%4d",j);  }  printf("\n");  }  getch();  } |  |

Output:

|  |  |  |
| --- | --- | --- |
|  | **26.a. To generate the pattern 3 2 1**  **3 2**  **1** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> |  |

|  |  |
| --- | --- |
| #include <conio.h>  void main()  {  int i, j, rows;  clrscr();  printf("Enter number of rows: "); scanf("%d",&rows);  for(i=rows; i>=1; i--)  {  for(j=i; j>=1; j--)  {  printf("%4d",j);  }  printf("\n");  }  getch();  } |  |

Output:

|  |  |  |
| --- | --- | --- |
|  | **28.a. To generate the pattern 1**  **2 3**  **4 5 6** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h>  #include <conio.h> |  |

|  |  |
| --- | --- |
| void main()  {  int i, j, ,k=1,rows;  clrscr();  printf("Enter number of rows: "); scanf("%d",&rows);  for(i=1; i<=rows; ++i)  {  for(j=1; j<=i; ++j)  {  printf("%4d",k++);  }  printf("\n");  }  getch();  } |  |

Output:

|  |  |  |
| --- | --- | --- |
|  | **29. To find the GCD and LCM of two numbers using functions.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main()  { |  |

|  |  |  |
| --- | --- | --- |
|  | int num1, num2, gcd, lcm; |  |
|  | int GCD(int,int); |
|  | clrscr(); |
|  | printf("Enter two numbers\n"); |
|  | scanf("%d%d", &num1, &num2); |
|  | gcd = GCD(num1, num2); |
|  | lcm = (num1 \* num2) / gcd; |
|  | printf("GCD of %d and %d = %d\n", |
|  | num1, num2, gcd); |
|  | printf("LCM of %d and %d = %d\n", |
|  | num1, num2, lcm); |
|  | getch(); |
| } |  |
| int | GCD(int x,int y) |
| { |  |
|  | while (x != y) |
|  | { |
|  | if ( x > y ) |
|  | x = x - y; |
|  | else |
|  | y = y - x; |
|  | } |
|  | return (x); |
| } |  |

Output:

|  |  |
| --- | --- |
| **32. To check whether a number is an Armstrong number using functions.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main()  {  int num, flag; |  |

|  |  |  |
| --- | --- | --- |
|  | int Armstrong(int); |  |
|  | clrscr(); |
|  | printf("Enter a numbers\n"); |
|  | scanf("%d", &num); |
|  | flag = Armstrong(num); |
|  | if (flag) |
|  | printf ("\n %d is an armstrong no", |
|  | num); |
|  | else |
|  | printf ("\n %d is not an armstrong no", |
|  | num); |
|  | getch(); |
| } |  |
| int | Armstrong(int n) |
| { |  |
|  | int num, sum = 0, rem = 0; |
|  | num = n; |
|  | sum = 0; |
|  | while (num != 0) |
|  | { |
|  | rem = num % 10; |
|  | sum = sum + rem \* rem \* rem; |
|  | num = num / 10; |
|  | } |
|  | if (sum == n) |
|  | return (1); |
|  | else |
|  | retrun (0); |
| } |  |

Output:

|  |  |
| --- | --- |
| **33. To calculate compound interest using functions.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main()  {  float prin, rate, ci; |  |

|  |  |  |
| --- | --- | --- |
|  | int tim; |  |
|  | float compint(float, int, float); |
|  | clrscr(); |
|  | printf("Enter prin., time and rate \n"); |
|  | scanf("%f%d%f", &prin, &tim, &rate); |
|  | ci = compint(prin, tim, rate); |
|  | printf(" Compound Interest = %f\n", ci); |
|  | getch(); |
| } |  |
| float | compint(float p, int t, float r) |
| { |  |
|  | int y=1; |
|  | float a=p; |
|  | while (y<=t) |
|  | { |
|  | a = a \* ( 1 + r / 100); |
|  | ++y; |
| } |  |
|  | return (a - p); |
| } |  |

Output:

|  |  |
| --- | --- |
| **34. To find the factorial of a number using recursion.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h> long int factorial(int n); void main()  {  int n; |  |

|  |  |
| --- | --- |
| clrscr();  printf("Enter a positive integer: "); scanf("%d", &n);  printf("Factorial of %d = %ld", n,  factorial(n));  getch();  }  long int factorial(int n)  {  if (n >= 1)  return (n\*factorial(n-1));  else  return (1);  } |  |

Output:

|  |  |
| --- | --- |
| **35. To find x to the power of y using recursion.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h> float power(float , int ); int main()  {  int y;  float x, p; |  |

|  |  |
| --- | --- |
| clrscr();  printf("\nEnter value of x and y: "); scanf("%f%d",&x,&y);  p = power(x,y);  printf("%f ^ %d = %f", x, y, p);  getch();  }  float power(float a, int b)  {  if (b == 0)  return (1); else if (b > 0)  return (x\*power(x, y-1));  else  return (1/x\*power(x, y+1));  } |  |

Output:

|  |  |
| --- | --- |
| **36. To generate “N” th Fibonacci series using recursion.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h> int fibo(int);  void main()  {  int num, result; |  |

|  |  |
| --- | --- |
| clrscr();  printf("\nEnter which number is to be  displayed: ");  scanf("%d", &num); if (num < 0)  printf("Fibonacci of negative number is  not possible.\n");  else  {  result = fibo(num);  printf("The %d number in fibonacci series is %d\n", num, result);  }  getch();  }  int fibo(int num)  {  if (num == 0)  return 0; else if (num == 1)  return 1;  else  return(fibo(num - 1) + fibo(num - 2));  } |  |

Output:

|  |  |
| --- | --- |
| **39. To find the largest and smallest of “N” numbers.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main()  {  int a[50],i,n,large,small; |  |

|  |  |  |
| --- | --- | --- |
|  | clrscr(); |  |
|  | printf("\nHow many elements:"); |
|  | scanf("%d",&n); |
|  | printf("Enter the Array:"); |
|  | for(i=0;i<n;++i) |
|  | scanf("%d",&a[i]); |
|  | large=a[0]; |
|  | small=a[0]; |
|  | for(i=1;i<n;++i) |
|  | { |
|  | if(a[i]>large) |
|  | large=a[i]; |
|  | if(a[i]<small) |
|  | small=a[i]; |
|  | } |
|  | printf("\nThe largest element is |
|  | %d",large); |
|  | printf("\nThe smallest element is |
|  | %d",small); |
|  | getch(); |
| } |  |

Output:

|  |  |
| --- | --- |
| **40. To perform Bubble sort.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main()  {  int count, temp, i, j, a[30]; |  |

|  |  |  |
| --- | --- | --- |
|  | clrscr(); |  |
|  | printf("\nHow many numbers: "); |
|  | scanf("%d",&count); |
|  | printf("\nEnter %d numbers: ",count); |
|  | for(i=0;i<count;i++) |
|  | scanf("%d",&number[i]); |
|  | for(i=1;i< count;i++) { |
|  | for(j=0;j<=i;j++) { |
|  | if(number[j]>number[j+1]){ |
|  | temp=number[j]; |
|  | number[j]=number[j+1]; |
|  | number[j+1]=temp; |
|  | } |
|  | } |
|  | } |
|  | printf("\nSorted elements: "); |
|  | for(i=0;i<count;i++) |
|  | printf(" %d",number[i]); |
|  | getch(); |
| } |  |

Output:

|  |  |
| --- | --- |
| **44. To perform Binary Search.** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h> void main()  {  clrscr();  int n, i, arr[50], search;  int first, last, middle, flag; |  |

|  |  |  |
| --- | --- | --- |
|  | printf("\nEnter number of elements :"); |  |
|  | scanf("%d",&n); |
|  | printf("\nEnter %d number :", n); |
|  | for (i=0; i<n; i++) |
|  | scanf("%d",&arr[i]); |
|  | printf("Enter a number to find :"); |
|  | scanf("%d", &search); |
|  | first = 0; |
|  | last = n-1; |
|  | flag = 0; |
|  | while ((first <= last) && !(flag)) |
|  | { middle = (first+last)/2; |
|  | if(arr[middle] == search) |
|  | { |
|  | flag = 1; |
|  | break; |
|  | } |
|  | else if(arr[middle] < search) |
|  | first = middle + 1; |
|  | else |
|  | last = middle - 1; |
|  | } |
|  | if (flag) |
|  | printf("\n%d found at location %d", |
|  | search, middle+1); |
|  | else |
|  | printf("\n%dNot found!",search); |
|  | getch(); |
| } |  |

Output:

|  |  |
| --- | --- |
| **45. To find the sum of two matrices.** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h> void main()  {  int a[10][10], b[10][10], c[10][10];  int i, j, m, n ;  clrscr(); |  |

|  |  |  |
| --- | --- | --- |
|  | printf("\nEnter the order of the |  |
|  | matrices"); |
|  | scanf("%d%d", &m, &n); |
|  | printf("\nEnter matrix 1 elements :"); |
|  | for(i=0; i<m; i++) |
|  | for(j=0; j<n; j++) |
|  | scanf("%d",&a[i][j]); |
|  | printf("Enter matrix 2 elements :"); |
|  | for(i=0; i<m; i++) |
|  | for(j=0; j<n; j++) |
|  | scanf("%d",&b[i][j]); |
|  | for(i=0; i<m; i++) |
|  | for(j=0; j<n; j++) |
|  | c[i][j]=a[i][j]+b[i][j]; |
|  | printf("\nThe new matrix will be :\n"); |
|  | for(i=0; i<m; i++) |
|  | { |
|  | for(j=0; j<3; j++) |
|  | printf("%d ",mat3[i][j]); |
|  | printf("\n"); |
|  | } |
|  | getch(); |
| } |  |

Output:

|  |  |  |
| --- | --- | --- |
|  | **46. To find the product of two matrices.** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h> void main()  {  int a[10][10], b[10][10], c[10][10];  int i, j, k,m, n, p, q;  clrscr(); |  |

|  |  |  |
| --- | --- | --- |
|  | printf("\nEnter the order of first |  |
|  | matrix"); |
|  | scanf("%d%d", &m, &n); |
|  | printf("\nEnter the order of second |
|  | matrix"); |
|  | scanf("%d%d", &p, &q); |
|  | if( n != p) |
|  | { |
|  | printf("\n Multiplication not |
|  | possible !"); |
|  | break; |
|  | } |
|  | else |
|  | { |
|  | printf("\nEnter matrix 1 |
|  | elements :"); |
|  | for(i=0; i<m; i++) |
|  | for(j=0; j<n; j++) |
|  | scanf("%d",&a[i][j]); |
|  | printf("\nEnter matrix 2 |
|  | elements :"); |
|  | for(i=0; i<p; i++) |
|  | for(j=0; j<q; j++) |
|  | scanf("%d",&b[i][j]); |
|  | printf("Multiplying two matrices...\n"); |
|  | for(i=0; i<m; i++) |
|  | { |
|  | for(j=0; j<q; j++) |
|  | { |
|  | c[i][j] = 0; |
|  | for(k=0; k<n; k++) |
|  | c[i][j] += + a[i][k] \* b[k][j]; |
|  | } |
|  | } |
|  | printf("\nResultant Matrix : \n"); |
|  | for(i=0; i<m; i++) |
|  | { |
|  | for(j=0; j<q; j++) |
|  | printf("%d ", c[i][j]); |
|  | printf("\n"); |
|  | } |
|  | getch(); |
| } |  |

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

Output:

|  |  |
| --- | --- |
| **47. To transpose a matrix.** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h> void main()  {  int a[10][10], b[10][10];  int i, j, m, n ;  clrscr();  printf("\nEnter the order of the matrix"); |  |

|  |  |  |
| --- | --- | --- |
|  | scanf("%d%d", &m, &n); |  |
|  | printf("\nEnter matrix elements :"); |
|  | for(i=0; i<m; i++) |
|  | for(j=0; j<n; j++) |
|  | scanf("%d",&a[i][j]); |
|  | printf("Transposing Array...\n"); |
|  | for(i=0; i<n; i++) |
|  | for(j=0; j<m; j++) |
|  | b[i][j]=a[j][i]; |
|  | printf("Transpose of the Matrix is :\n"); |
|  | for(i=0; i<n; i++) |
|  | { |
|  | for(j=0; j<m; j++) |
|  | printf("%d ",b[i][j]); |
|  | printf("\n"); |
|  | } |
|  | getch(); |
| } |  |

Output:

|  |  |
| --- | --- |
| **50. To find the frequency of occurrence of a character in a string.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <string.h> #include <conio.h>  void main()  {  char str[100], ch; int i, count = 0;  printf("Enter a String \n"); |  |

|  |  |
| --- | --- |
| gets(str);  printf("Enter a character \n"); ch = getchar();  for(i = 0; str[i] != ‘\0’; i++)  if ( str[i] == ch) ++count;  printf(“%c occurs %d times in %s \n”,  ch,count,str);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **51. To reverse a string .** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <string.h> #include <conio.h>  void main()  {  char str[100], rstr[100]; int i, L;  printf("Enter a String \n"); |  |

|  |  |
| --- | --- |
| gets(str);  L= strlen(str);  for(i = L-1; i>=0 ; i--)  rstr[L-i+1] = str[i];  rstr[l]='\0';  printf(“Original String = %s \n”, str); printf(“Reverse String = %s \n”, rstr);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **52. To copy the contents of one string to another.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <string.h> #include <conio.h>  void main()  {  char str[100], cstr[100]; int i, l;  printf("Enter a String \n"); |  |

|  |  |
| --- | --- |
| gets(str);  l= strlen(str);  for(i = 0; i<l ; i++)  cstr[i] = str[i];  cstr[i]='\0';  printf(“Original String = %s \n”, str); printf(“Copied String = %s \n”, cstr);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **56. To swap value of two variables using pointers.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main ()  {  int a,b;  void swap(int \*, int \*);  clrscr();  printf("Enter two numbers \n"); scanf("%d%d",&a,&b); |  |

|  |  |
| --- | --- |
| printf("Before swapping\n"); printf("a = %d , b = %d \n", a, b );  swap(&a, &b);  printf("After swapping\n"); printf("a = %d , b = %d \n", a, b ); getch();  }  **/\* function definition to swap the values \*/**  void swap(int \*x, int \*y)  {  int temp;  temp = \*x;  \*x = \*y;  \*y = temp;  return;  } |  |

Output:

|  |  |  |
| --- | --- | --- |
|  | **57. To calculate the area and circumference of a circle using pointers.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  void main ()  {  float r,a,c;  void calc(float , float \*, float \*);  clrscr();  printf("Enter the radius \n"); scanf("%f",&r); |  |

|  |  |
| --- | --- |
| clac(r, &a, &c);  printf("Radius = %f \n", r); printf("Area = %f \n", a ); printf("Circumference = %f \n", c ); getch();  }  void calc(float x, float \*y , float \*z)  {  \*y = 3.14 \* x \*x;  \*z = 2 \* 3.14 \* x;  return;  } |  |

Output:

|  |  |
| --- | --- |
| **58. To sort a list of numbers using pointers.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h> void main()  {  int n, a[100], \*p;  int , i,j,temp;  clrscr();  printf("\nHow Many Numbers: "); scanf("%d",&n);  printf(“Enter %d numbers \n”,n); |  |

|  |  |
| --- | --- |
| for(i=0;i<n;i++)  scanf("%d",&a[i]);  p=a; for(i=0;i<n;i++) for(j=0;j<n;j++)  {  if(\*(p+i)<\*(p+j))  {  temp=\*(p+i);  \*(p+i)=\*(p+j);  \*(p+j)=temp;  }  }  printf("\nSorted Numbers Are:\n"); for(i=0;i<n;i++)  printf("%d \t",a[i]);  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **60. To create a student database using structures.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h> #include <string.h>  struct student  {  char name[50]; int USN;  float marks;  }; |  |

|  |  |
| --- | --- |
| void main()  {  int i;  struct student s[10];  clrscr();  printf("Enter information of 10  students:\n");  for(i=0; i<10; ++i)  {  s[i].roll = i+1;  printf("\nFor USN %d,\n",s[i].roll); printf("Enter name: "); gets(s[i].name);  printf("Enter Total marks: "); scanf("%f",&s[i].marks); printf("\n");  }  printf("Displaying Student  Information:\n\n");  for(i=0; i<10; ++i)  {  printf("\nUSN : %d\n",i+1); printf("Name: ");  puts(s[i].name);  printf("Total Marks: %.1f",s[i].marks); printf("\n");  }  getch();  } |  |

Output:

|  |  |
| --- | --- |
| **61. To create and use an Union.** | |
| **Program** | **Algorithm and Flowchart** |
| #include <stdio.h> #include <conio.h>  union test  {  int x, y;  };  void main()  {  union test t; |  |

|  |  |
| --- | --- |
| clrscr();  t.x = 2;  printf(" After assigning value for x \n"); printf(" x = %d y = %d \n",t.x, t.y);  t.y = 10;  printf(" After assigning value for y \n"); printf(" x = %d y = %d \n",t.x, t.y); getch();  } |  |

Output:

|  |  |
| --- | --- |
| **62. To create a file for employee data.** | |
| **Program** | **Algorithm and Flowchart** |
| #include<stdio.h> #include<conio.h>  void main()  {  FILE \*fptr ;  int i, n, empno ;  float bpay, allow, ded ; char name[10] ;  clrscr() ;  fptr = fopen("EMPLOYEE.DAT", "w") ; |  |

|  |  |
| --- | --- |
| printf("Enter number of employees : ") ; scanf("%d", &n) ;  for(i = 0 ; i < n ; i++)  {  printf("\nEnter employee number : ") ; scanf("%d", &empno) ;  printf("\nEnter the name : ") ; scanf("%s", name) ;  printf("\nEnter the basic pay, allowances  & deductions : ") ; scanf("%f%f%f", &bpay, &allow, &ded) ; fprintf(fptr, "%d %s %f %f %f \n",  empno,name,bpay,allow,ded);  }  fclose(fptr) ;  fptr = fopen("EMPLOYEE.DAT", "r") ; printf("\nEmp. No.Name\t\t Bpay\t\t  Allow\t\t Ded\t\t Npay\n\n"); for(i = 0 ; i < n ; i++)  {  fscanf(fptr,"%d%s%f%f%f\n", &empno,name,&bpay,&allow,&ded);  printf("%d \t %s \t %.2f \t %.2f \t %.2f  \t %.2f \n", empno, name, bpay, allow, ded,  bpay + allow - ded) ;  }  fclose(fptr) ;  getch() ;  } |  |

Output: