**SET – 9**

Q.52. Write a program using enum to design a calendar.

#include<stdio.h>

int main()

{

char a1[12][50]={"January", "February","March", "April","May","June","July","August","September","October","November","December"};

enum date {x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12,x13,x14,x15,x16,x17,x18,x19,x20,x21,x22,x23,x24,x25,x26,x27,x28,x29,x30,x31};

char c1[7][6]={"Sun","Mon","Tue","Wed","Thur","Fri","Sat"};

int i=0;

int j=0;

int k=0;

int m=0;

while(i<12)

{

printf("\t%s\n",a1[i]);

while(1)

{

if(j==7)

{

printf("\n");

j=0;

break;

}

printf("%s\t",c1[j]);

j++;

}

while(m!=0)

{

printf("\t");

m--;

}

if(i==0||i==2||i==4||i==6||i==7||i==9||i==11)

{

enum date r=x1;

while(1)

{

printf("%d\t",r+1);

m++;

if(k==6)

{

printf("\n");

m=0;

k=0;

}

else

k++;

if((r+1)==31)

{

break;

}

r++;

}

}

else if(i==1)

{

enum date r=x1;

while(1)

{

printf("%d\t",r+1);

m++;

if(k==6)

{

printf("\n");

m=0;

k=0;

}

else

k++;

if((r+1)==28)

{

break;

}

r++;

}

}

else

{

enum date r=x1;

while(1)

{

printf("%d\t",r+1);

m++;

if(k==6)

{

printf("\n");

m=0;

k=0;

}

else

k++;

if((r+1)==30)

{

break;

}

r++;

}

}

printf("\n");

i++;

}

}

Q53. Write a program using structure to prepare record maintainance system of all the students in the batch. Also use Union in the same program somewhere. Enter the student records and process them later.

#include<stdio.h>

int main()

{

struct student{

char name[50];

int r\_no;

int choice;

};

union student2{

long int m\_no;

int l\_no;

};

printf("Enter No. of Students\n");

int n;

scanf("%d",&n);

struct student s[n];

union student2 s1[n];

int i=0;

while(i<n)

{

printf("For Student %d:\n",(i+1));

printf("Enter Name:\n");

scanf("%s",s[i].name);

printf("Enter Roll No\n");

scanf("%d",&s[i].r\_no);

printf("Enter 1 for Landline No. or 2 for Mobile No.\n");

scanf("%d",&s[i].choice);

switch(s[i].choice)

{

case 1:printf("Enter Landline No.--\n");

scanf("%d",&s1[i].l\_no);

break;

case 2:printf("Enter Mobile No.--\n");

scanf("%ld",&s1[i].m\_no);

break;

}

printf("\n");

i++;

}

i=0;

printf("Details entered are:\n");

while(i<n)

{

printf("For Student ---%d:\n",(i+1));

printf("Name: %s\n",s[i].name);

printf("Roll No: %d\n",s[i].r\_no);

switch(s[i].choice)

{

case 1:printf("Landline No.: %d\n",s1[i].l\_no);

break;

case 2:printf("Mobile No.: %ld\n",s1[i].m\_no);

break;

}

printf("\n");

i++;

}

}

Q53-53. Write the same program as given in question 52 using file programming. Take a file with given roll numbers and names of the students of your batch. Use all the file operations that are possible to be done.

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int main()

{

FILE \*fp=fopen.txt","r"); //enter the address of the file

int temp;

char temp2[50];

char temp3[50];

printf("Roll No.\tName\n");

while(!feof(fp))

{

fscanf(fp,"%d",&temp);

fscanf(fp,"%s",temp2);

printf("%d\t\t%s\n",temp,temp2);

strcpy(temp2,temp3);

}

printf("\n");

}

**SET – 8**

42. Program to create, initialize, assign and access a pointer variable.

Top of Form

|  |  |
| --- | --- |
|  |  |
|  | #include <stdio.h> |
|  |  |
|  | int main() |
|  | { |
|  | int a = 6; |
|  | int \*ptr; |
|  | ptr = &a; |
|  |  |
|  | printf("Value at a is %d\n", a); |
|  | printf("Address of a is %u\n", ptr); |
|  | printf("Value of a accessed by pointer is %d\n", \*ptr); |
|  | }  Q 43 Program to swap two numbers using pointers.   |  | | --- | | #include <stdio.h> | |  | void swap(int\*,int\*); | |  | int main() | |  | { | |  | int a,b; | |  | printf("Enter two numbers: \n"); | |  | scanf("%d%d", &a,&b); | |  | printf("Before a = %d and Before b = %d\n", a,b); | |  | swap(&a,&b); | |  | printf("After a = %d and After b = %d\n",a,b); | |  | } | |  |  | |  | void swap(int \*x,int \*y) | |  | { | |  | int temp; | |  | temp = \*x; | |  | \*x=\*y; | |  | \*y=temp; | |  |  | |  | }  Q44- Program to change the value of constant integer using pointers.  #include<stdio.h>  int main()  {  const int a=7;  printf("Enter Value of a: ");  scanf("%d",&a);  int\* ptr=&a;  printf("Original Value of a: %d\n",\*ptr);  \*ptr = 10;  printf("New Value assigned to a: %d\n",\*ptr);  }  Q45 Program to print a string using pointer.  #include<stdio.h>  int main()  {  char a[50];  char\* ptr=a;  printf("Enter a string\n");  scanf("%s",ptr);  printf("%s\n",ptr);  }  Q 46. Program to count vowels and consonants in a string using pointer.  #include<stdio.h>  #include<string.h>  int main()  {  char a[50];  char\* ptr=a;  printf("Enter String");  scanf("%s",ptr);  int count\_v=0;  while(\*ptr!='\0')  {  if(\*ptr=='a'||\*ptr=='e'||\*ptr=='i'||\*ptr=='o'||\*ptr=='u'||\*ptr=='A'||\*ptr=='E'||\*ptr=='I'||\*ptr=='O'||\*ptr=='U')  count\_v++;  ptr++;  }ptr=a;  printf("No. of vowels is: %d\n",count\_v);  printf("No. of consonants is: %lu\n",(strlen(ptr)-count\_v));  }  Q 47. Program to read array elements and print with addresses.  #include <stdio.h>  main()  {  int array[20] = {1,2,3,4,5,6,7},i;  int \*ptr;  ptr = array;    for(i=0;i<7;i++){  printf("Element: %d Address: %u\n",array[i],ptr);  ptr++;  }  }  Q 48. Program to read and print student details using structure pointer, demonstrate example of structure with pointer.  #include<stdio.h>  int main()  {  struct student{  int r\_no;  char name[20];  int class;  };  struct student x;  struct student\* ptr=&x;  printf("Enter Details:\n");  printf("Enter Roll No\n");  scanf("%d",&((\*ptr).r\_no));  printf("Enter Name:\n");  scanf("%s",(\*ptr).name);  printf("Enter Class:\n");  scanf("%d",&((\*ptr).class));  printf("Details entered are:\n");  printf("Roll No.: %d\n", (\*ptr).r\_no);  printf("Name: %s\n", (\*ptr).name);  printf("Class: %d\n", (\*ptr).class);  }  Q 49. Program to print size of different types of pointer variables.  #include <stdio.h>  main()  {  int i=2, \*ptri;  float f=2.0, \*ptrf;  char c='7',\*ptrc;    ptri = &i;  ptrf = &f;  ptrc = &c;    printf("Size of integer pointer = %d bytes\n ", sizeof(\*ptri));  printf("Size of float pointer = %d bytes\n ", sizeof(\*ptrf));  printf("Size of character pointer = %d byte\n ", sizeof(\*ptrc));    }  Q 50. Program to demonstrate example of double pointer (pointer to pointer).  #include<stdio.h>  int main()  {  int x=50;  int\* ptr=&x;  int \*\*ptr2=&ptr;  printf("Address stored in double pointer is: %p\n",ptr2);  printf("Address of single pointer is: %p\n",&ptr);  printf("Address stored in single pointer is: %p\n",ptr);  printf("Address stored in single pointer by dereferencing double pointer is: %p\n",\*ptr2);  printf("Address of x is: %p\n",&x);  printf("Value stored in x is: %d\n",x);  printf("Value stored in x by dereferencing double pointer is: %d \n",\*\*ptr2);  printf("Address of double pointer is: %p\n",&ptr2);  }  Q 51. Program to demonstrate example of array of pointers.  #include<stdio.h>  void main()  {  int (\*arr)[10];  int i=0;  for(;i<10;i++)  {  scanf("%i",\*(arr[i]));  }  for(;i<10;i++)  {  printf("Address is: %p\tValue is: %d\n",arr[i],\*(arr[i]));  }  }  **SET – 7**  37. Write a program to display the fibonacci series upto a number that is taken as input and passed to the user-defined function.  #include <stdio.h>  main()  {  int n,x;  int fibo(int n);  printf("Enter the limit\n");  scanf("%d", &n);  x=fibo(n);    }  int fibo(n)  {  int a=0,b=1,s=0;    while(s <= n){  a=b;  b=s;  printf("%d ",s);  s=a+b;      }    }    38. Write a user-defined funtion to evaluate factorial of a number taken as input.  #include <stdio.h>  int main()  {  int n,x;  int fact(int n);  printf("Enter the number\n");  scanf("%d", &n);  x=fact(n);    }  int fact(n)  {  int f=1;    while(n>0){  f=f\*n;  n--;        }  printf("Factorial is : %d",f);  }  39. Take any two pairs of coordinates as input which form two different circles of the same radius 10 units. Find whether the two circles intersect; if they do, find the area of intersection; otherwise, display "doesn't intersect".  #include<stdio.h>  #include<math.h>  int main()  {  float coor[2][2];  printf("Enter 1st Coordinate:\n");  scanf("%f%f",&coor[0][0],&coor[0][1]);  printf("Enter 2nd Coordinate:\n");  scanf("%f%f",&coor[1][0],&coor[1][1]);  float dist=sqrt(pow(coor[1][0]-coor[0][0],2)+pow(coor[1][1]-coor[0][1],2));  if(dist>20.0)  printf("Doesn't Intersect\n");  else if(dist<20.0)  {  printf("Intersects at 2 Points\n");  float angle=acos(dist/(2.0\*10.0));  float aoi=(1/2.0)\*pow(10.0,2)\*(angle-sin(angle));  printf("Area of Intersection is: %f\n",aoi);  }  else if(dist==20.0)  printf("Intersects at one point only.\nArea of intersection = 0\n");  }    40. Take any two 4 x 4 matrices such as A and B, multiply them and store the resultant matrix in matrix C. Display C. Take user input as A or B to perform addition of C and A or B. Show the result of addition again. Follow matrix multiplication rules.    #include<stdio.h>  void initialize(int (\*a)[4]);  void multiplication(int (\*c)[4],int \*m1,int \*m2,int i,int j);  void display(int (\*c)[4]);  int main()  {  int A[4][4],B[4][4],C[4][4],m1[4],m2[4];  printf("Values Of A\n");  initialize(A);  display(A);  printf("Values Of B\n");  initialize(B);  display(B);  int i,j,k;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  {  for(k=0;k<4;k++)  {  m2[k]=B[k][j];  m1[k]=A[i][k];  }  multiplication(C,m1,m2,i,j);  }  }  display(C);  }  void multiplication(int (\*c)[4],int \*m1,int \*m2,int i,int j)  {  int k=0,product=0;  for(;k<4;k++)  product+=((\*(m1+k))\*(\*(m2+k)));  (\*((\*(c+i))+j))=product;  }  void initialize(int (\*a)[4])  {  printf("Enter Values\n");  int i=0;  for(;i<4;i++)  scanf("%d %d %d %d",(\*(a+i)),(\*(a+i)+1),(\*(a+i)+2),(\*(a+i)+3));  }  void display(int (\*c)[4])  {  int i,j;  for(i=0;i<4;i++)  {  for(j=0;j<4;j++)  printf("%d\t",\*((\*(c+i))+j));  printf("\n");  }  }  41. Take any continuous function f(x) and input two coordinates for the upper and lower limit. Take another input coordinate to check whether it satsfies the Lagrange's mean value theorem or not. f(x) and all the input should be taken in such a manner so that the theorem is once satisfied and once not satisfied.  #include<stdio.h>  #include<math.h>  int main()  {  int n;  float x1,x2;  printf("Enter Degree of the Polynomial: \n");  scanf("%d",&n);  float coeff[n+1];  int i=0;  while(i<=n)  {  printf("Enter coefficient of x^%d:\n",i);  scanf("%f",&coeff[i]);  i++;  }  i=0;  printf("Enter x1 and x2\n");  scanf("%f %f",&x1,&x2);  float sumx1=0;  float sumx2=0;  while(i<=n)  {  sumx1+=coeff[i]\*pow(x1,i);  sumx2+=coeff[i]\*pow(x2,i);  i++;  }  float mean=(sumx2-sumx1)/(x2-x1);  printf("Enter value of x\n");  float x;  scanf("%f",&x);  i=1;  float sumx3=0;  while(i<=n)  {  sumx3+=coeff[i]\*i\*pow(x,i-1);  i++;  }  if((round(mean\*100)/100.0)==(round(sumx3\*100)/100.0))  {  printf("Mean value theorem is satisfied\n");  }  else  {  printf("Mean value theorem not satisfied\n");  }  }  SET – 6  34. Write a program that takes any two pairs of input for x- and y-coordinate in cartesian coordinate system and suggests the output as the third coordinate that makes the triangle an equilateral triangle.  #include<stdio.h>  #include<math.h>  int main()  {  float x1,x2,y1,y2;  scanf("%f%f%f%f",&x1,&y1,&x2,&y2);  if(y1!=y2){  float x=(x1+x2)/2.0;  float y=(y1+y2)/2.0;  float m=(y2-y1)/(x2-x1);  float m2=-1/m;  float i,j;  for(i=0;;i+=0.000001)  {  j=(m2\*(i-x))+y;  float c1=sqrt(pow(x2-i,2)+pow(y2-j,2));  float c2=sqrt(pow(x1-i,2)+pow(y1-j,2));  float c3=sqrt(pow(x2-x1,2)+pow(y2-y1,2));  int p=(int)(round(c1\*100));  int q=(int)(round(c2\*100));  int r=(int)(round(c3\*100));  if((p==q)&&(p==r))  {  printf("The coordinate is:%f %f\n",i,j);  goto outer;  }  }      outer:;  }  else{  float i,j;  i=(x1+x2)/2;  for(j=0;;j+=0.000001)  {  float c1=sqrt(pow(x2-i,2)+pow(y2-j,2));  float c2=sqrt(pow(x1-i,2)+pow(y1-j,2));  float c3=sqrt(pow(x2-x1,2)+pow(y2-y1,2));  int p=(int)(round(c1\*100));  int q=(int)(round(c2\*100));  int r=(int)(round(c3\*100));  if((p==q)&&(p==r))  {  printf("The coordinate is:%f %f\n",i,j);  goto outer2;  }  }  outer2:;  }  }    35. Write a program that takes three numbers as input from the user and checks whether any combination of these form a Pythagorean triplet or not. A Pythagorean triplet is the one in which the sum of squares of two numbers is equal to the square of the third number.  #include<math.h>  #include<stdio.h>  main()  {  int a,b,c;  printf("Enter Three Numbers:\n");  scanf("%d %d %d", &a,&b,&c);  if(a==sqrt((b\*b)+(c\*c))){  printf("Pythagorean Triplet Exists");  }  else if(b==sqrt((a\*a)+(c\*c))){  printf("Pythagorean Triplet Exists");  }  else if(c==sqrt((a\*a)+(b\*b))){  printf("Pythagorean Triplet Exists");  }  else  printf("Pythagorean Triplet Does not Exists");  }  36. Write a program using the function getchar() that inputs a rational numbers and operators and then does that operation on them and prints it using putchar(). scanf or printf should not be used. ASCII values may be used to convert characters to integers.  #include <stdio.h>  main()  {  int a,b,c,temp=0;  printf("Enter the numbers with the operator:\n");  a=getchar();  b=getchar();  c=getchar();  switch(b){  case '+' :  temp = (a + c)-48;  putchar(temp);  break;  case '-' :  temp = (a - c);  printf("%d",temp); ////////  break;  case '\*' :  temp = (a \* c)-48;  putchar(temp);  break;  }  }  **SET -5**  30. Input a range from user and print all the narcissistic number in that range. (A number is called narcissistic if each of its digits raised to the power of the number of digits  equals the number.)  #include <stdio.h>  #include <math.h>  main()  {  int a,n1,n2,d=0,m,sum=0;  printf("Program To Find All the Narcissistic number within a range\n");  printf("Enter two numbers: \n");  scanf("%d",&n1);  scanf("%d",&n2);  while(n1<=n2){  m=n1;  while(m>0){  m = m/10;  d++;  }  m=n1;  while(n1>0){  a = n1%10;  sum = sum + pow(a,d);  n1 = n1 / 10;  }  n1=m;  n1++;    if(m == sum)  printf("%d is Narcissistic\n",m);      sum=0;  d=0;  }    }  31. Print the following (You may use a single nested loop if you can't do it one loop)  #include<stdio.h>  main()  {  int rows,sp,st,i=0;    for(rows=1;rows<=9;){  for(sp=1;sp<=rows;sp++){  printf(" ");  }  for(st=1;st<=10-rows;st++){  printf("\* ");  }  if(rows == 9)  i = 1;    if(i == 0)  rows++;  else{  rows--;  if(rows == 0)  break;  }    printf("\n");  }  }    32. Store your name in a one dimensional array and print it from the array character-wise using a loop.  #include <stdio.h>  main()  {  int i;  char n[6] = {'D', 'H', 'I', 'R', 'A', 'J'};    for(i=0;i<=5;i++){  printf("%c\n",n[i]);  }  }  33. Using 2 dimensional array, develop a playable tic-tac-toe. Print the result after every move. Declare a winner or draw as the game ends.  #include<stdio.h>  int main()  {  char a[3][3]={{' ',' ',' '},{' ',' ',' '},{' ',' ',' '}};  char b[3];  int m,i,x,y,j,k;  char l;  printf("Tic Tac Toe\n");  for(j=0;j<3;j++)  {  for(k=0;k<3;k++)  {  if(k==0||k==1)  printf(" %c |",a[j][k]);  else  printf(" %c ",a[j][k]);  }  if(j!=2)  printf("\n---+---+---");  printf("\n");  }  for(i=0;i<9;i++)  {  if(i%2==0)  {  m=1;  printf("Player %d\n",m);  }  else  {  m=2;  printf("Player %d\n",m);  }  printf("Enter Position\n");  scanf("%d%d",&x,&y);  if(m==1)  a[x-1][y-1]='X';  if(m==2)  a[x-1][y-1]='O';  printf("Tic Tac Toe\n");  for(j=0;j<3;j++)  {  for(k=0;k<3;k++)  {  if(k!=2)  printf(" %c |",a[j][k]);  else  printf(" %c ",a[j][k]);  }  if(j!=2)  printf("\n---+---+---");  printf("\n");  }  for(j=0;j<3;j++)  {  for(k=0;k<3;k++)  {  b[k]=a[j][k];  }  if((b[0]==b[1])&&(b[1]==b[2])&&(b[1]!=' '))  {  printf("Player %d wins\n",m);  goto outer;  }  }  for(j=0;j<3;j++)  {  for(k=0;k<3;k++)  {  b[k]=a[k][j];  }  if((b[0]==b[1])&&(b[1]==b[2])&&(b[1]!=' '))  {  printf("Player %d wins\n",m);  goto outer;  }  }  if((a[0][0]==a[1][1])&&(a[2][2]==a[1][1])&&(a[1][1]!=' '))  {  printf("Player %d wins\n",m);  goto outer;  }  if((a[0][2]==a[1][1])&&(a[1][1]==a[2][0])&&(a[1][1]!=' '))  {  printf("Player %d wins\n",m);  goto outer;  }  }  printf("Game is Draw\n");  outer: ;  }  **SET -4**  25. The outstanding balance on a home loan Rs. 8,00000. Each month a payment of Rs 30000 is made which includes both interest and principal repayment of the car loan. The yearly interest is calculated as 11% of the outstanding balance of the loan for the first year and 14% for the rest. After the interest is deducted the remaining part of the payment is used to payoff the loan. Using this information, write a C program that produces a table indicating the beginning monthly balance, the interest payment, the principal payment, and the remaining loan balance after each payment is made. Your output should resemble and complete the entries in the following table until the outstanding loan balance is zero.  #include<stdio.h>  main()  {  float interest,pa=800000,rem,b,new\_interest,c,d,new\_interest1;  int i;  interest = 0.11\*pa;  b = pa+interest;  printf("Interest for First Year = %.2f\n",interest);  printf("Principle Amount For 1st year is %.2f\n",b);  printf("--------------First Year-------------\n");  printf("Month Monthly Balance Rem Balance\n");  for(i=1;i<=12;i++){  printf("%d\t ", i);  printf("%.2f\t ", b);  rem = b - (30000);  printf("%.2f \n", rem);  b = rem;  }  new\_interest = 0.14 \* b;  c = b + new\_interest;  printf("Interest for Second Year = %.2f\n", new\_interest);  printf("Principle Amount For 2nd year is %.2f\n", c);  printf("--------------Second Year-------------\n");    printf("Month Monthly Balance Rem Balance\n");  for(i=1;i<=12;i++){  printf("%d\t ", i);  printf("%.2f\t ", c);  rem = c - (30000);  printf("%.2f \n", rem);  c = rem;  }    new\_interest1 = 0.14 \* c;  d = new\_interest1 + c;    printf("Interest for Third Year = %.2f\n", new\_interest1);  printf("Principle Amount For 3rd year is %.2f\n", d);  printf("--------------Third Year-------------\n");    printf("Month Monthly Balance Rem Balance\n");    for(i=1;i<=12;i++){  printf("%d\t ", i);  printf("%.2f\t ", d);  rem = d - (30000);    if(rem>=0) {  printf("%.2f \n", rem);    }    else  break;    d = rem;  }  printf("0000.00");      }  Q 26. Write a program to check whether a given number is an Armstrong number or not. An Armstrong number is the one in which the sum of cubes of its digit is equal to the number itself. Your program should take a number as input and output whether the given number is an Armstrong number or not.  #include <stdio.h>  #include <stdlib.h>  main()  {  int n,rem,cube,sum = 0,temp,num;  printf("Enter the Number:");  scanf("%d", &n);  temp = n;  while(n > 0){  rem= n % 10;  cube = rem \* rem \* rem; //For calculating cubes  sum = sum + cube;  n = n / 10; //For separate digits  }  n = temp;  if (n == sum){  printf("%d is an Armstrong Number", n);  }  else{  printf("%d is not an Armstrong Number", n);  }  }  Q 27. Write a program to assign a grade according to the marks received in an exam. Make use of the switch statement. Your program should take the marks as input from the user and print the appropriate grade.  #include <stdio.h>  main()  {  int marks;  char grade;    printf("Enter Marks:");  scanf("%d", &marks);    grade = ((marks >= 80) ? 'A' : (marks >= 65) ? 'B' : (marks>=50) ? 'C' : (marks>=30) ? 'D' : 'F');    switch(grade){  case 'A' :  printf("Grade is A\n");  break;  case 'B' :  printf("Grade is B\n");  break;  case 'C' :  printf("Grade is C\n");  break;  case 'D' :  printf("Grade is D\n");  break;  default :  printf("Grade is F");  }      return 0;  }  Q28. Take two numbers a and b as input from the user and print the sum of the squares of all the odd numbers between a and b (including a and b)  #include <stdio.h>  #include <stdlib.h>  int main()  {  int a, b, sqr, sum;  printf("Enter first number: \n");  scanf("%d", &a);  printf("Enter second number: \n");  scanf("%d", &b);  sum = 0;  for (;a <= b; a++){  if (a % 2 != 0){  sqr = a \* a;  sum = sum + sqr;  }  else{  continue;  }  }  printf("%d\n", sum);  }  Q29 There are 100 students in a class. There is a event A: a student knows C programming, event B: a student knows Fortran programming. Take suitable inputs for P(A or B), P(A and B) and another variable of your choice. Write a program to find the outputs of P(B|A), P(A|B), P(not B| not A), P(not A| not B) using Bayes theorem.  #include <stdio.h>  main()  {  float pa,pb,pAorB,pAandB,pBgivenA,pAgivenB,pnotBgivennotA,pnotAgivennotB;    printf("There are 100 students in a class.\n");  printf("Event A = a student knows C programming and Event B = ");  printf("a student knows Fortran programming\n");  printf("Enter P(A):\n");  scanf("%f", &pa);  printf("Enter P(A or B):\n");  scanf("%f", &pAorB);  printf("Enter P(A and B):\n");  scanf("%f", &pAandB);    pb = pAorB + pAandB - pa;    printf("Therefore P(B) = %f\n", pb);    pBgivenA = pAandB / pa;  pAgivenB = pAandB / pb;  pnotBgivennotA = (1 - pAorB) / (1 - pa);  pnotAgivennotB = (1 - pAorB) / (1 - pb);      printf("P(B | A) is = %f\n", pBgivenA);  printf("P(A | B) is = %f\n", pAgivenB);  printf("P(notB | notA)= %f \n", pnotBgivennotA);  printf("P(notA | notB)= %f \n", pnotAgivennotB);    }  **SET - 3**  **//Q-21** Find the summation of an series for the first 30 numbers: 2.n^2 + 3.n + 5.  #include <stdio.h>  int main()  {  int i=0,sum=0;  for(;i<=30;i++)  {  sum=sum+(2\*i\*i)+(3\*i)+5;    } printf("sum= %d \n", sum);    i=0,sum=0;  while(i<=30)  {  sum=sum+(2\*i\*i)+(3\*i)+5;    i++;  }  printf("sum= %d \n", sum);  i=0,sum=0;  do  {  sum=sum+(2\*i\*i)+(3\*i)+5;    i++;  }  while(i<=30);  printf("sum= %d \n", sum);  return 0;  }  **//Q-22** Print all the numbers between 1001 to 2001 that are divisible by 7.  #include <stdio.h>  int main()  {  int i=1001;  do  {  if(i%7==0)  printf("%d \n",i);  i++;  }  while(i<=2001);    i=1001;  while(i<=2001)  {  if(i%7==0)  printf("%d \n",i);  i++;    }  i=1001;  for(i=1001;i<=2001;i++)  {  if(i%7==0)    printf(" %d \n", i);  }  return 0;  }  **//Q-**23 Print all the leap years between 1901 to 1999.  #include <stdio.h>  int main()  {  int year=1901;  printf("while loop\n");  while(year<=1999)  {  if(year%4==0)  printf("the leap years is %d \n", year);  year++;  }  printf("for loop\n");  year=1901;  for(;year<=1999;year++)  {  if(year%4==0)  printf("The leap year is %d \n", year);  }  printf("do while loop\n");  year=1901;  do  { if(year%4==0)  printf("the leap year is %d \n",year);  year++;  }  while(year<=1999);  return 0;  }  **//Q-**24Write any program using loop showing the use of break and continue statement.  #include <stdio.h>  int main()  {  int i = 1;  for (; i <= 15; i++)  {  printf ("The entered no are: %d\n", i);  if (i == 8)  break;  }  i = 1;  for (; i <= 15; i++)  {  printf ("The entered no are: %d\n", i);  if (i == 8)  continue;  }  return 0;  **SET – 2**  /**Q-13** Use scanf ("%[^\n]%\*c", name); take the following input:  Hey how are you? and print the same.  #include <stdio.h>  int main()  {  char name[25];  scanf ("%[^\n]%\*c", name);  printf("THE STRING IS: %s \n", name);  return 0;  }  Q-14 Use the following format specifiers:  %3.2f, %-4d, %1s, %15.8s, %+6d, %#x, %#f  #include <stdio.h>  int main()  {  float a=2.56;  int b=333;  char c[25] ="India is my country";  int d=20.5;  printf("%3.2f \n", a);  printf("%-4d \n", b);  printf("%1s \n", c);  printf("%15.8s\n", c);  printf("%+6d \n", b);  printf("%#f \n", a);  printf("%#x \n", d);    }  Q-15 Take any integer and check whether its prime  #include <stdio.h>  int main()  {  int n, i, flag = 0;  printf("Enter a positive integer: ");  scanf("%d",&n);  for(i=2; i<=n/2; ++i)  {  // condition for nonprime number  if(n%i==0)  {  flag=1;  break;  }  }  if (flag==0)  printf("%d is a prime number.",n);  else  printf("%d is not a prime number.",n);    return 0;  }  16. Take suitable value for each of the following data types: short integer, long integer, unsigned long integer, signed long integer, double, char, long double.Display and justify the output of each with the sizeof(value) operator.  #include<stdio.h>  int main()  {  short int a=127; long int b=2147483647; unsigned long int c=4294967295; signed long int d=2147483647; double e=0.786553456564455654433;  long double f=0.7234686426484686847784236478;char g=127;  printf("Size of Short Int:%ld byte \n",sizeof(a));  printf("Size of Long Integer:%ld byte\n",sizeof(b));  printf("Size of Unsigned Long Integer:%lu byte\n",sizeof(c));  printf("Size of Signed Long Integer:%ld byte\n",sizeof(d));  printf("Size of Double:%ld byte \n",sizeof(e));  printf("Size of Long Double:%ld byte\n",sizeof(f));  printf("Size of Char:%ld byte\n",sizeof(g));  }  **Q-18** Write a program to differentiate %d and %u output for a any integer variable.  #include <stdio.h>  int main()  {  int a;  unsigned int b;  scanf("%d", &a);  scanf("%u", &b );  printf("the no is: %d \n", a);  printf("the no. is %u" ,b);    return 0;  }  **Q-**19 Write a program using ternary operator for condition checking.  #include <stdio.h>  int main()  {  float n;  printf("enter the no.:");  scanf("%f ", &n);  (n>0) ? printf("the no. is positive") : printf("the no. is negative");  return 0;  }    **Q 20**. Take any integer i and justify the results of:  i++, ++i, ++i, i++, --i, i--, i--, --i in the given order.  #include <stdio.h>  int main()  {  int i = 4;  printf("%d %d %d %d %d %d %d %d", i++,++i,++i,i++,--i,i--,i--,--i);    }  **SET-1**  Q-1-Write a C program that compute sum of two short integers and display the result.  #include <stdio.h>  int main()  {  short int a;  short int b;  int c;  printf("the first number is :");  scanf("%hd ", &a);  printf("%hd \n", a);  printf("the second number is :");  scanf("%hd ", &b);  printf("%hd \n", b);  c=a+b;  printf("the sum is : %d" , c);      return 0;  }  **Q-**2 Write a C program that perform the addition of an unsigned integer constant and a long  integer constant and stored the result in a long integer variable.  #include <stdio.h>  int main()  {  unsigned const int A;  long const int B;  long int c;  printf("the first number is:");  scanf("%d ", &A);  printf(" %d \n ", A);  printf("the second number is:");  scanf("%d ",&B);  printf("%d \n", B);  c=A+B;  printf("the sum is: %d", c);  return 0;  **}**  **Q-**3 Write a program that calculates the area of a circle by taking radius as an input from keyboard.  #include <stdio.h>  int main()  {  float r,area;  printf("the radius is: ");  scanf("%f ", &r);  printf("%f \n", r);  area=3.14 \* r \* r;  printf("the area is: %f" , area);  return 0;  }  Q-4 Let Principal = P, Rate = R% per annum and Time = T years. Then simple interest = (PxRxT)/100. Write a C program that compute the simple interest with user given input of P, R, and T.  \*/#include <stdio.h>  int main()  {  float p,r,t,interest;  printf("The principal amount is:");  scanf("%f " ,&p);  printf("%f \n" ,p);  printf("The rate of interest is:");  scanf("%f " ,&r);  printf("%f \n" ,r);  printf("The time is:");  scanf("%f " ,&t);  printf("%f \n" ,t);  interest=p\*r\*t/100;  printf("the simple interest is: %f",interest);    }  **Q-**5Print the following:  "Hello",'How are you?'  #include <stdio.h>  int main()  {  printf(" \"Hello\", 'how are you?'");  return 0;  }  **Q-6** Perform summation, subtraction, multiplication and division with integer numbers only.  #include <stdio.h>  int main()  {  float a, b, sum, sub, mult , division;  printf("Enter the two numbers: ");  scanf("%f %f",&a,&b);  printf("%f %f \n", a,b);  sum=a+b;  printf("the sum is: %f \n", sum);  sub=a-b;  printf("the subtraction is: %f \n", sub);  mult=a\*b;  printf("the multiplication is: %f \n", mult);  division=a/b;  printf("the division is: %f", division);  return 0;  }  **Q-**7 Perform summation, subtraction, multiplication and division with integer numbers only.  #include <stdio.h>  int main()  {  float a, b, sum, sub, mult , division;  printf("Enter the two numbers: ");  scanf("%f %f",&a,&b);  printf("%f %f \n", a,b);  sum=a+b;  printf("the sum is: %f \n", sum);  sub=a-b;  printf("the subtraction is: %f \n", sub);  mult=a\*b;  printf("the multiplication is: %f \n", mult);  division=a/b;  printf("the division is: %f", division);  return 0;  **}**  **Q-**8 Print octal and hexadecimal values of an integer.  #include <stdio.h>  int main()  {  int a=144;  printf("the hexadecimal number is :%x \n", a);  printf("the octal number is :%o", a);  return 0;  }  **Q-9** Provide an input of a string and display.  #include <stdio.h>  int main()  {  char str[20];  gets(str);  printf("The string is %s ", str);  return 0;  }  **Q-10** Print size of the data types using sizeof operator.  #include <stdio.h>  int main()  {    char a ='A';  int b =120;  float c =123.0f;  double d =1222.90;  char str[] ="Hello";    printf("\nSize of a: %d",sizeof(a));  printf("\nSize of b: %d",sizeof(b));  printf("\nSize of c: %d",sizeof(c));  printf("\nSize of d: %d",sizeof(d));  printf("\nSize of str: %d",sizeof(str));  return 0;  }  **Q-11**Take any two integers and perform all the following operators as used: OR, NOT, AND, XOR, >>, <<  #include <stdio.h>  int main()  {  unsigned int a=12, b=13;  int c=0;      c= a & b;  printf("a & b= %d \n",c);  c=a | b;  printf("a | b= %d \n",c);  c=~a;  printf("a = %d \n",c);  c=a^b;  printf("a ^ b= %d \n",c);  c = a << 2;  printf("the value of left shift oper= %d \n", c );  c = a >> 2;  printf("the vlaue of right shift oper= %d\n", c );            return 0;  }  **Q-12 A**dd two short integers and store the result in a long variable. Print the results case 1: by shifting 8 positions right and case 2: by shifting 8 positions left**.**  #include <stdio.h>  int main()  {  short int a,b;  long int c,d,e;  printf("the first no. is= ");  scanf("%hd ",&a);  printf("%hd \n", a);  printf("the second no. is= ");  scanf("%hd ",&b);  printf("%hd \n", b);  c=a+b;  printf("the sum of the no. is= %ld \n",c);  d= c << 7;  printf("the left shift operation : %ld \n", d);  e= c >> 7;  printf("the right shift operation : %ld \n", e);  return 0;  }  **CLASS TEST 2 –**  1. Write a program which shows the usage of all the followings: (2 marks)  fopen, fclose, feof, fscanf, fprintf, fgets, fputs, fgetc, fputc  #include<stdio.h>  #include<stdlib.h>  int main()  {  char c,b[200],b1[200];  FILE \*fp=fopen(".txt","r");    while(1)  {  c = fgetc(fp);  if(feof(fp))  {  break;  }  printf("%c", c);  }  printf("\n");  fclose(fp);  FILE \*fp2=fopen(".txt","r");  fscanf(fp2,"%s",b);  printf("First Word is:%s\n",b);  fclose(fp2);  FILE \*fp3=fopen(".txt","r");  fgets(b1,7,fp3);  printf("First 6 characters are:%s\n",b1);  fclose(fp3);  FILE \*fp4=fopen(".txt","w");  fprintf(fp4,"%s"," Hello World.");  fclose(fp4);  FILE \*fp5=fopen(".txt","w");  fputs("system programming language.", fp4);  fclose(fp5);  FILE \*fp6=fopen("/.txt","r");  int d=fgetc(fp6);  printf("First Character :%c\n",d);  fclose(fp6);  return(0);  }    2. Use malloc() to allocate space for any three variables of any datatype. Print the contents and addresses of the variables. Use free() to free the memory locations. Print the contents of the variables again.  #include<stdio.h>  #include<stdlib.h>  int main()  {  int\* ptr;  ptr = (int \*) malloc(10 \* sizeof(int));  int\* ptr2=ptr;  int i=0;  while(i<10)  {  \*ptr=i;  i++;  ptr++;  }  i=0;  ptr=ptr2;  printf("Not Removing\n");  while(i<10)  {  printf("Value :%d\n",\*ptr);  printf("Address :%p\n",ptr);  i++;  ptr++;    }  printf("Removing\n");  free(ptr2);  ptr=ptr2;  i=0;  while(i<10)  {  printf("Value is:%d\n",\*ptr);  printf("Address is:%p\n",ptr);  i++;  ptr++;  }  }  3. Allocate space for an array using calloc().  \*/  #include <stdio.h>  #include <stdlib.h>  int main()  {  int n=15,i,\*p;  p=(int\*)calloc(n, sizeof(int));  if(p == NULL)  {  printf("cannot be allocated\n");  }  else{  printf("Elements of array : \n");  for(i=0;i<n;i++)  {  printf("%d\n",\*(p+i));  }  }  return 0;  }  4. Write a simple program to differentiate call by value and call by references.  #include <stdio.h>  void swapcov(int,int);  void swapcor(int\*,int\*);  int main()  {  int a,b;  printf("Enter two numbers: \n");  scanf("%d %d",&a,&b);  swapcov(a,b);  printf("The value of a= %d and b=%d using Call by value\n",a,b);  swapcor(&a,&b);  printf("The value of a=%d and b=%d using Call by reference\n", a,b);  }  void swapcov(int x,int y)  {  int temp;  temp = x;  x=y;  y=temp;  }  void swapcor(int \*x,int\*y)  {  int temp;  temp = \*x;  \*x=\*y;  \*y=temp;  }  5. Write a program to show any three string operations using strings.h.  #include <stdio.h>  #include <string.h>  main()  {  char src[50];  int result;  char s1[10]="Hello",s2[10]="World";  printf("--For String concatenate--\n");  printf("Enter a string not more than 50 characters\n");  gets(src);  strcat(src, "This string is concatenated");  puts(src);    printf("--For String copy--\n");  strcpy(src,"The string is copied\n");  puts(src);    result = strcmp(s1, s2);    if (result == 0)  printf("Strings are the same\n");    if (result < 0)  printf("Second string is less than the first\n");  return 0;  }  6. Write a program to create two blank files, write your name in one, copy it's contents to another blank flie and delete the first file. Each operation should be performed after any user input.  #include<stdio.h>  #include<stdlib.h>  int main()  {  char temp[20],temp2[20],temp3[20];  FILE \*fp=fopen(".txt","w");  printf("Enter (Yes) to print name to file:\n");  scanf("%s",temp);  fprintf(fp,"Dhiraj");  fclose(fp);  char ch;  printf("Enter (Yes) to start copying\n");  scanf("%s",temp2);  FILE \*fp2=fopen(".txt","w");  FILE \*fp3=fopen(".txt","r");  while((ch=fgetc(fp3))!=EOF)  {  printf("%c",ch);  fputc(ch,fp2);  }  fclose(fp2);  printf("Enter (Yes) to remove file\n");  scanf("%s",temp3);  remove(".txt");  fclose(fp3);  }  7. Use file pointer to search a sub-string within a large string.  #include <stdio.h>  void main()  {  char string[1000], substring[1000];  int position,len, a=0;  printf("Input a string under 1000 characters\n");  scanf("%s",string);  printf("Enter the position and length of sub-string\n");  scanf("%d%d", &position, &len);  while (a<len) {  substring[a]=string[position+a-1];  a++;  }  substring[a]='\0';  printf("substring is \"%s\"\n", substring);  }  8. Run a system command of Operating system and store the result of the command in a file (Use argc, \*argv[] and any other necessary operations).  #include<stdio.h>  int main()  {  system("touch dhiraj.txt"); // To create a file name dhiraj.txt  return 0;  }  9. Write a program to use strerror() and perror() and show some error messages.  #include <stdio.h>  #include <errno.h>  #include <string.h>    int main ()  {  int b;  printf("Enter Value of a\n");  scanf("%d",&b);  printf("Value of errno: %d\n", errno);  printf("Message using strerror() : %s\n", strerror(errno));  perror("Message from perror");  return 0;  }  10. Write a simple strcuture with typedef to display your name and roll number.  #include <stdio.h>  #include <string.h>    typedef struct {  char name[50];  char rollno[50];  int semester;    }students;    int main( ) {  students amit;  printf("Enter your name:\n");  scanf("%s",amit.name);    printf("Enter your Roll No:\n");  scanf("%s",amit.rollno);    printf("Enter your Semester\n");  scanf("%d",&amit.semester);  printf("%s,%s,%d", amit.name,amit.rollno,amit.semester);    }  11. Write a program to use any two functions from the following C libraries: (marks: 3)  C Library - <ctype.h>  C Library - <errno.h>  C Library - <float.h>  C Library - <limits.h>  C Library - <locale.h>  C Library - <math.h>  C Library - <setjmp.h>  C Library - <signal.h>  C Library - <stdarg.h>  C Library - <stddef.h>  C Library - <stdio.h>  C Library - <stdlib.h>  C Library - <string.h>  C Library - <time.h>  #include <ctype.h>  #include <errno.h>  #include <float.h>  #include <limits.h>  #include <locale.h>  #include <math.h>  #include <setjmp.h>  #include <signal.h>  #include <stdarg.h>  #include <stddef.h>  #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <time.h>  #include <unistd.h>  main()  {  //#include <ctype.h>  printf("\nCTYPE.H");  printf("\n FUNCTION: ISALPHANUMERIC()");  int var1='D';  if( isalnum(var1) ) {  printf("\nvar1 = %c is alphanumeric", var1 );  } else {  printf("\nvar1 = %c is not alphanumeric", var1 );  }  printf("\n FUNCTION: ISDIGIT()");  if( isdigit(var1) ) {  printf("\nvar1 = %c is a digit\n", var1 );  } else {  printf("\nvar1 = %c is not a digit\n", var1 );    int b;  printf("Enter Value of a\n");  scanf("%d",&b);  printf("Value of errno: %d\n", errno);  printf("Message using strerror() : %s\n", strerror(errno));  perror("Message from perror");  }  printf("\n LIMITS.H");  printf("\nThe maximum value of SIGNED CHAR using SCHAR\_MAX= %d", SCHAR\_MAX);  printf("\nThe maximum value of UNSIGNED CHAR using UCHAR\_MAX= %d", UCHAR\_MAX);    printf("\n FLOAT.H");  printf("\nThe maximum value of float using FLT\_MAX = %.10e", FLT\_MAX);  printf("\nThe minimum value of float using FLT\_MIN = %.10e", FLT\_MIN);  //math.h  printf("\n\nMATH.H");  printf("\n FUNCTION:SQRT()");  printf("\n Square root of 51: %f ",sqrt(51));  printf("\n FUNCTION:POW()");  printf("\n 53 to the power of 2.37: %f",pow(53, 2.37));  printf("\n\nSTRING.H");  char str1[10]="AB";  char str2[]="EF";  printf("\n FUNCTION:STRCMP()");  printf("\n str1=AB AND str2=EF, comparing them using strcmp, RESULT= %d",strcmp(str1,str2));  printf("\n FUNCTION:STRCAT()");  printf("\n str1=AB AND str2=EF, concating them using strcat, RESULT= %s",strcat(str1,str2));  printf("For stdio.h");  int a;  printf("Enter a number:\n");  scanf("%d",&a);  printf("value at a is %d",a);  printf("\n\nTIME.H");  printf("\n FUNCTION:DIFFTIME()");  time\_t start\_t, end\_t;  double diff\_t;  printf("\n Starting of the program...");  time(&start\_t);  printf("\n Sleeping for 10 seconds...");  sleep(5);  time(&end\_t);  printf("\n Execution time = %f", difftime(end\_t, start\_t));  printf("\n FUNCTION:TIME()");  time\_t seconds;  seconds = time(NULL);  printf("\nHours since January 1, 1970 = %ld", seconds/3600);  printf("\n END of the program...");    }  12. Write to program to use all the following storage classes: (marks: 1)  auto  register  static  extern  #include<stdio.h>  int r;  extern void use\_extern();  void test();  int main()  {  register int c = 10;  test();  test();  test();  auto int i = 1;  {  auto int i = 2;  {  auto int i = 3;  printf ( "\n%d\t", i);  }  printf ( "%d\t", i);  }  printf( "%d\n", i);  r = 5;  use\_extern();  }  void test()  {  static int a = 0;  a = a + 1;  printf("%d\t",a);  }  The extern file  #include <stdio.h>  extern int r;    void use\_extern(void) {  printf("count is %d\n", r);  } | |  |  | |

Bottom of Form