AIM

To write a C-program to convert a given number into binary number

ALGORITHM

STEP1:Start the program execution

STEP2:Read the decimal number

STEP3:Perform number mod 2 to find the reminder and save the reminder in sum.

STEP4:Divide the number by 10 to get the quotion

STEP5:Print the binary number in sum

STEP6:Stop the program execution

PROGRAM

```
*/
#include <stdio.h>
#include <stdlib.h>
// this program convert no into binary

void main()
{
long int n,i=1,sum=0,r;
system("clear");
printf("\nEnter the decimal number:");
scanf("%ld",&n);
while(n>0)
{
r=n%2;
sum=sum+(r*i);
i=i*10;
n=n/2;
}
printf("\n Equivalent binary number is %ld",sum);
getchar();
}
/* this program convert no into binary
```

OUTPUT

Enter the decimal no:10

Equivalent binary nois 1010

RESULT

Thus the C-program had been wreitten ,executed sussesfully to perform decimal to binary conversion*/

```
Enter the decimal number:4

 Equivalent binary number is 100
```

```
Enter the decimal number:45

 Equivalent binary number is 101101
```

AIM : To write a C program to calculate the Simple and Compound intrest.

```
PROCEDURE
STEP1:start the program
STEP2:declare the values p,n,ch,r
STEP3:input the values for p,n,r,ch
STEP4:get the choice from user weather to calcualte simple or compound intrest
STEP5:claculate the print the value of simple intrest  or compound intrest
STEP6:stop the program
CODING*/
#include<stdio.h>
#include<math.h>
#include <stdlib.h>
// sometimes #ivclude<coinio.h> dont work in linux
// sometimes clrscr does not work
//sometimes getch does not work
// so we use system("clear")and getchar
int main()
{
int p,n,ch;
float r;
system("clear");
printf("\n Enter the choice:\n1.Simple intrest \n2.Comound intrest: \nchoice: ");
scanf("%d",&ch);
printf("\nEnter the value of amount , number of years and rate of intrest:");
scanf("%d%d%f",&p,&n,&r);
switch(ch)
{
case 1: printf("\n The simple intrest is = %f", (p * n * r) / 100);
break;

case 2: printf("\n The compound intrest is = %f",(p * pow((1 +(r /100)),n)-p));
break;
default:printf("\nEnter the correct choice:");
break;
}
return 0;
}
/* OUTPUT
Enter the choice:
  1. simple intrest
  2. compound intrest
choice:1
Enter the value of amount ,number of years and Rate of interest:
100 2 2
The simple  intrest is =400
RESULT
Thus the C program has been written, executed succesfully to calculate the simple
interest and compound*/
```

```
 Enter the choice:
1.Simple intrest
2.Comound intrest:
choice: 1

Enter the value of amount , number of years and rate of intrest:1000
4
15

 The simple intrest is = 600.000000
```

```
 Enter the choice:
1.Simple intrest
2.Comound intrest:
choice: 2

Enter the value of amount , number of years and rate of intrest:1000
4
2

 The compound intrest is = 82.432160
```

AIM:To write a C program to find the area and peremeter ofcircle,triangle,rectangle and square

```
PROCEDURE
STEP1:Start the program execution
STEP2:Declare the variable ch,a,b,c,h,r
STEP3:Get the choice weather the  1.circle  2.square 3.rectangle 4.triangle
STEP4:If ch=1 read r and compute area  and perimeter of circle
STEP5:Print the area and the perimeter of circle
STEP6:If ch=2 read a and copmute area perimeter of square
STEP7:Print the  area and perimeter of square
STEP8:If ch=3 read b,h and compute area and perimeter of reactangle
STEP9:Print the area and the perimeter of rectangle
STEP10:If ch=4 read a,b,c and compute area  and perimeter of triangle
STEP11:Print the area and perimeter of triangle
STEP12:Stop the program execution
CODING
*/
#include<stdio.h>
#include<stdlib.h>
void main()
{
int ch ,b,h,a,c;
float r;
system("clear");
printf(" Enter the choice:\n1.circle\n2.square\n3.rectangle\n4.triangle\nchoice:");
scanf("%d",&ch);
switch (ch)
{
case 1: printf("\nenter the radius of the circle:");/* constant-expression */
   scanf("%f",&r);
   printf("\narea of circle=%f",3.14 * r * r);
   printf("\nperimeter of circle=%f",2*3.14 * r);
    /* code */
   break;
case 2: printf("\nEnter the side of the square:");
    scanf("%d",&a);
    printf("\nArea of the square id %d",a*a);
    printf("\nPerimeter of square=%d",4*a);

    break;
case 3: printf("\nEnter the breadth and height of rectangle:");
   scanf("%d%d",&b,&h);
   printf("\nArea of the rectangle:%d",b*h);
   printf("\nPerimeter of the rectangle:%d",2*(b+h));
   break;
case 4:printf("\nEnter the breadth and height of triangle :");
     scanf("%d%d",&b,&h);
     printf("\nEnter the three side of triangle:");
     scanf("%d%d%d",&a,&b,&c);
     printf("\nArea of triangle:%f",(float)(0.5*b*h));
     printf("\nPerimeter of triangle:%f",(float)(a+b+c));
     break;
default:printf("\nEnter the correct choice:");
break;
}
getchar;
}
/*OUTPUT
```

```
Enter the  choice :
  1.Circle
  2.Square
  3.Rectangle
  4.Triangle
Chice 1
Enter the radius of the circle:2
Area of ciccle=12.56
Perimeter of circle=12.56
*/
```

```
 Enter the choice:
1.circle
2.square
3.rectangle
4.triangle
choice:1

enter the radius of the circle:15

area of circle=706.500000
perimeter of circle=94.200000
```

```
 Enter the choice:
1.circle
2.square
3.rectangle
4.triangle
choice:2

Enter the side of the square:43

Area of the square id 1849
Perimeter of square=172
```

```
 Enter the choice:
1.circle
2.square
3.rectangle
4.triangle
choice:3

Enter the breadth and height of rectangle:13
23

Area of the rectangle:299
Perimeter of the rectangle:72
```

```
 Enter the choice:
1.circle
2.square
3.rectangle
4.triangle
choice:4

Enter the breadth and height of triangle :12
14

Enter the three side of triangle:12
13
14

Area of triangle:91.000000
Perimeter of triangle:39.000000
```

```c
#include<stdio.h>
int main()
{
int arr[10],i,n,sum=0;
printf("\n enter the number of elements in the arry ");
// reading the size of the array
scanf("%d",&n);
printf("\n enter the values for the element of the array ");
for(i=0;i<n;i++)
{
   printf("\n enter the element %d ",i+1);
   scanf("%d",&arr[i]);
}
for (i=0;i<n;i++)
{
    sum=sum+arr[i];
}
printf("\n printing the array elements .....");
printf("\n the values for the elements of the array");
for (i=0;i<n;i++)
{
    printf("\n the elements %d :%d",i+1,arr[i]);
}
printf("\n printing the  sum of array ...");
printf("%d",sum);
}
```

```
enter the number of elements in the arry 3

enter the values for the element of the array
enter the element 1 3

enter the element 2 5

enter the element 3 2

printing the array elements .....
the values for the elements of the array
the elements 1 :3
the elements 2 :5
the elements 3 :2
printing the  sum of array ...10
```

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

struct student {
    int id;
    char name[30];
    float percentage; // data
};#include <stdio.h>
#include <string.h>
#include <stdlib.h>

struct student {
    int id;
    char name[30];
    float percentage; // data
};

int main() {
    int i;
    system("clear"); // This will clear the terminal screen (works on Unix/Linux)

    struct student record[3]; // Array of 3 student structures

    // 1st student's record
    record[0].id = 1;
    strcpy(record[0].name, "raju");
    record[0].percentage = 86.5;

    // 2nd student's record
    record[1].id = 2;
    strcpy(record[1].name, "surender"); // Fixed: should be record[1] instead of
record[2]
    record[1].percentage = 90.5;

    // 3rd student's record
    record[2].id = 3;
    strcpy(record[2].name, "thiyagu");
    record[2].percentage = 81.5;

    // Displaying the records
    for (i = 0; i < 3; i++) {
        printf("\nRecord of student %d:\n", i + 1);
        printf("Id: %d\n", record[i].id);
        printf("Name: %s\n", record[i].name);
        printf("Percentage: %.2f\n\n", record[i].percentage); // Changed to %.2f
for better formatting
    }

    return 0; // Added return statement for main
}

int main() {
    int i;
    system("clear"); // This will clear the terminal screen (works on Unix/Linux)

    struct student record[3]; // Array of 3 student structures

    // 1st student's record
```

```
    record[0].id = 1;
    strcpy(record[0].name, "raju");
    record[0].percentage = 86.5;

    // 2nd student's record
    record[1].id = 2;
    strcpy(record[1].name, "surender"); // Fixed: should be record[1] instead of
record[2]
    record[1].percentage = 90.5;

    // 3rd student's record
    record[2].id = 3;
    strcpy(record[2].name, "thiyagu");
    record[2].percentage = 81.5;

    // Displaying the records
    for (i = 0; i < 3; i++) {
        printf("\nRecord of student %d:\n", i + 1);
        printf("Id: %d\n", record[i].id);
        printf("Name: %s\n", record[i].name);
        printf("Percentage: %.2f\n\n", record[i].percentage); // Changed to %.2f
for better formatting
    }

    return 0; // Added return statement for main
}
```

```
Record of student 1:
Id: 1
Name: raju
Percentage: 86.50


Record of student 2:
Id: 2
Name: surender
Percentage: 90.50


Record of student 3:
Id: 3
Name: thiyagu
Percentage: 81.50
```

```
/*
AIM
To write a C program to design a simple menu-driven calculator
ALGORITHM
STEP1.Start the program execution
STEP2.Declare 2 variable a and b ,ch,c
STEP3.Get choice 1.addition,2.subtraction,3.multiplaction,4.dividion,5.modulation
STEP4.If ch=1,perform addition and print the result
STEP5.If ch=2,perform subtration and print the result.
STEP6.If ch=3,perform multiplaction and print the result.
STEP7.If ch=4,perform dividion and print the result.
STEP8.If ch=5,perform modulation and print the results
STEP9.If choice is not in them print"Enter correct choice"
STEP10.Stop thr program execution
PROGRAM*/
 #include<stdio.h>
 void  main()
 {
 int a,b,c,ch;
 printf("\n Enter the  values:");
 scanf("%d%d",&a,&b);
 printf("\n Enter the choice:
\n1.Addition\n2.Subtraction\n3.Multiplication\n4.division\n5.Modulo\nchoice:");
 scanf("%d",&ch);
 switch (ch)

 {
 case 1: printf("\n Addition of %d+%d is %d",a,b,a+b);
 break;
 case 2: printf("\n subtraction of %d-%d is %d",a,b,a-b);
 break;
 case 3: printf("\n Multiplicaion of %d*%d is %d ",a,b,a*b);
 break;
 case 4:if(b==0)
 printf("\nDiviion is not possiible");
 else
 printf("\n Division of %d/%d is %f",a,b,(float)a/b);
 break;
 case 5: printf("\n modulus  of %d%%%d is %d ",a,b,a%b);
 break;
 default: printf("\n Enter correct choice");
 break;
 }
 getchar();
 }
```

```
 Enter the  values:13
14

 Enter the choice:
1.Addition
2.Subtraction
3.Multiplication
4.division
5.Modulo
choice:1

 Addition of 13+14 is 27
```

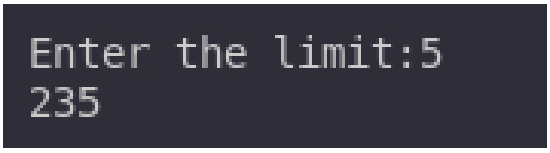
```
 Enter the  values:16
74

 Enter the choice:
1.Addition
2.Subtraction
3.Multiplication
4.division
5.Modulo
choice:2

 subtraction of 16-74 is -58
```

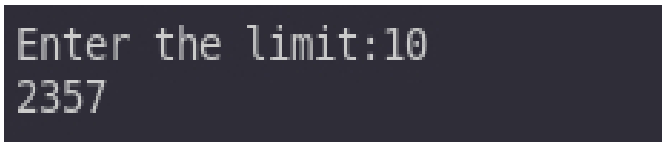
```
 Enter the  values:18
86

 Enter the choice:
1.Addition
2.Subtraction
3.Multiplication
4.division
5.Modulo
choice:3

 Multiplicaion of 18*86 is 1548
```

```
 Enter the  values:100
5

 Enter the choice:
1.Addition
2.Subtraction
3.Multiplication
4.division
5.Modulo
choice:4

 Division of 100/5 is 20.000000
```

```
 Enter the  values:123
12

 Enter the choice:
1.Addition
2.Subtraction
3.Multiplication
4.division
5.Modulo
choice:5

 modulus  of 123%12 is 3
```

```
#include<stdio.h>
void main()
{
int n,i,j;
printf("\nEnter the limit:");
scanf("%d",&n);
for (i=1;i <=n;i++)
{
for (j=2;j<i;j++)
{
if(i%j==0)
break;
}
if(i==j)
printf("%d",i);
}
getchar();
}
```





```c
#include <stdio.h>

long int factorial(long int);
int main()
{
long int no,fact;
//system("clear");
printf("\n enter a number...");
scanf("%ld",&no);
fact=factorial(no);
printf("\n the factorial of the given number id :%ld",fact);
getchar;
}
long int factorial(long int n)
{
int i;
long int fact=1;
for(i = 1;i <= n; i++){
  fact=fact*i;
}
return fact;
}
```

```
enter a number...5

the factorial of the given number id :120
```

```
enter a number...20

the factorial of the given number id :-2102132736
```

```
/*
AIM
To write a C-program to swap two numbers using function by
(a) Pass by value
(b) Pass by reference
ALGORITHM
STEP1.Start the program execution
STEP2.Read a and b
STEP3.Print a and b values before swaping
STEP4.Call function swap and pass 2 variable addresses
STEP5.Function swap catches those addresses and swaps the addresses using a tempary
varaible
STEP6.After swaping,print the values of a and b
STEP7.Stop the program execution
PROGRAM
*/
#include<stdio.h>
#include<stdlib.h>
void swap(int*,int*);

void main()
{
 int a,b;
 system("clear");
 printf("\nEnter two numbers:");
 scanf("%d%d",&a,&b);
 printf("\nCALL BY VALUE\n");
 printf("\nBefore swaping\n\n a=%d and b=%d",a,b);
 swap(&a,&b);
 printf("\nAfter swaping\n\n a=%d and b=%d",a,b);
 getchar();
 }
 void swap(int *a, int *b)
 {
 int t;
 t=*a;
 *a=*b;
 *b=t;

 }
 /*
 OUTPUT
 Enter two number : 10 20
 CALL BY VALUE
 Before swaping
 a=10 and b=20
 After swaping
 a=20 and b=1
 RESULT
 Thus,the C-program had been written,executed successfully to swap
  two values using functions by call value  and call by reference
*/
```

```
Enter two numbers:15
45

CALL BY VALUE

Before swaping

 a=15 and b=45
After swaping

 a=45 and b=15
```

```
/*
AIM
To write a C-program to print the Fibnacci series using functions.
ALGORITHM
STEP1.Start the program execution
STEP2.Read the Finbonacci range()
STEP3.Call function Fibonacci numbers
STEP4.Print the Fibonacci numbers
STEP5.Stop the program execution
PROGRAM*/
#include<stdio.h>
#include<stdlib.h>
void fib(int);
void main()
{
int n;
system("clear");
printf("\nEnter the limit:");
scanf("%d",&n);
fib(n);
getchar();
}
void fib(int n)
{
int i,a=0,b=1,c;
printf("\nFibnocci series are:");
printf("\n\n%d\t%d",a,b);
for ( i = 3; i < n; i++)
{
    c=a+b;
    a=b;
    b=c;
    printf("\t%d",c);
}

}
/*
OUPUT
Enter the the limit:5
Fibonacci series are
0 1 1 3 5 */
```

```
Enter the limit:28

Fibnocci series are:

0       1       1       2       3       5       8       13      21      34      55      89      144     233     377   6
10      987     1597    2584    4181    6765    10946   17711   28657   46368   75025   121393
```

```
Enter the limit:19

Fibnocci series are:

0       1       1       2       3       5       8       13      21      34      55      89      144     233     377   6
10      987     1597
```

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

struct stu {
    char name[100];
    int maths, phy, chem, rank;
    float cutoff;
};

int main() {
    struct stu s[100], temp;
    int i, j, n;

    printf("\n\nNo. of Students: ");
    scanf("%d", &n);
    getchar(); // consume newline after number input

    for (i = 0; i < n; i++) {
        printf("\nEnter Student %d Name: ", i + 1);
        fgets(s[i].name, sizeof(s[i].name), stdin);
        s[i].name[strcspn(s[i].name, "\n")] = 0;  // remove trailing newline

        printf("Maths Mark: ");
        scanf("%d", &s[i].maths);

        printf("Physics Mark: ");
        scanf("%d", &s[i].phy);

        printf("Chemistry Mark: ");
        scanf("%d", &s[i].chem);

        s[i].cutoff = (s[i].maths / 2.0) + (s[i].phy / 4.0) + (s[i].chem / 4.0);
        getchar(); // to consume newline after int
    }

    // Sort in descending order of cutoff
    for (i = 0; i < n - 1; i++) {
        for (j = i + 1; j < n; j++) {
            if (s[i].cutoff < s[j].cutoff) {
                temp = s[i];
                s[i] = s[j];
                s[j] = temp;
            }
        }
    }

    // Assign ranks
    for (i = 0; i < n; i++) {
        s[i].rank = i + 1;
    }

    printf("\n\n%-5s %-15s %-7s %-8s %-10s %-7s\n", "Rank", "Name", "Maths",
"Physics", "Chemistry", "Cutoff");
    for (i = 0; i < n; i++) {
        printf("%-5d %-15s %-7d %-8d %-10d %-7.2f\n", s[i].rank, s[i].name,
s[i].maths, s[i].phy, s[i].chem, s[i].cutoff);
    }
```

```
    return 0;
}
```

```
No. of Students: 3

Enter Student 1 Name: Hiler
Maths Mark: 123
Physics Mark: 453
Chemistry Mark: 541

Enter Student 2 Name: Trump
Maths Mark: 42
Physics Mark: 134
Chemistry Mark: 421

Enter Student 3 Name: Putin
Maths Mark: 332
Physics Mark: 532
Chemistry Mark: 213


Rank  Name            Maths   Physics  Chemistry  Cutoff
1     Putin           332     532      213        352.25
2     Hiler           123     453      541        310.00
3     Trump           42      134      421        159.75
```

```c
#include<stdio.h>
struct employee
{
    int age ;
    float percent ;
    char *name;
};
int main(){
    FILE *fp ;
    struct employee emp[]={
        {25,65.5,"Ravi"},
        {21,75.5,"Roshan"},
        {24,60.5,"Reena"}
    };
    char *string ;
    fp = fopen("/home/sagar/Documents/hacking-/C language project
/student.txt","w");
    for (int i= 0; i < 3; i++){
        fprintf(fp,"%d %f %s\n",emp[i].age,emp[i].percent,emp[i].name);

    }
    fclose(fp);
}
```

```
25 65.500000 Ravi
21 75.500000 Roshan
24 60.500000 Reena
```

```
#include<stdio.h>
//Selection sort implementation
void selectionSort(int arr[],int n){
   for (int i = 0; i < n; i++)
   {
        int min =i;
        for (int j = i+1; j < n; j++)
        {
            if(arr[j]<arr[min])
             min =j;
        }
        if(min!=i){
            int temp = arr[min];
            arr[min]=arr[i];
            arr[i]=temp;
        }

   }


}
int main(){
 int arr[]={2,6,1,5,3,4};

 int n = sizeof(arr)/sizeof(arr[0]);
 //Perform Selection Sort
 selectionSort(arr,n);
 printf("\nSortted array:");
 for(int i=0;i<n;i++)
   printf("%d ",arr[i]);
 return 0;
}
```

```
Sortted array:1 2 3 4 5 6
```

```
#include<stdio.h>
#include<stdlib.h>
void main()
{
   int c,pos;
   FILE *fpl;
   system("clear");
   fpl=fopen("/home/sagar/Documents/hacking-/C language project
/student.txt","r+");//the file has something to write in it
   printf("\nReading from the file ...\n");
   while(!feof(fpl))
   {
       c=fgetc(fpl);
       printf("%c",c);

   }
rewind(fpl);//position the pointer to the begging of the file
c=fgetc(fpl);
printf("\nThe  first character from the file : % c",c);
pos=ftell(fpl);//return th ecurrent position of the file pointer
printf("\nThe position of the file pointer :%d",pos);
fseek(fpl,10,SEEK_SET);//Moves the files pointer to the 10th byte  from the
beginning
pos=ftell(fpl);
c=fgetc(fpl);
printf("\nThe character at %d is %c  (from the beginning)",pos,c);
fseek(fpl,-5,SEEK_CUR);// moves the file pointer back to the 5th byte
//from the current position
pos=ftell(fpl);
c=fgetc(fpl);
printf("\n The character at %d id : %c",pos,c);
fseek(fpl,12,SEEK_CUR);//moves the file pointer fordward to the 12 th
//byte from the current position
pos=ftell(fpl);
c=fgetc(fpl);
printf("\nThe character at %d is :%c",pos,c);
getchar();
}
```