

TABLE OF CONTENTS

1) Write a C Program to Find the factorial of a given number. Input at run time.	2
2) Write a C Program to Find the sum of digits of a given number.	3
3) Write a C Program to reverse the digits of a given number.	5
4) Write a C program to convert a character. Lower to upper and upper to lower.	6
5) Write a C program to print multiplication tables from 10 to 20.	7
6) Write a C program to print first 100 prime numbers.	8
7) Write a C program to print Armstrong Numbers between 1 to 500.	9
8) Write a C program to print the binary of a given number (+ve or -ve numbers).	10
9) Write a C program to reverse the bits of a given number.	11
10) Write a C program to implement a Calculator using switch.	13
11) Write a C program to find the complement of a given number & then print it.	15
12) Write a C program to find the 2's complement of a given number and then print it.	17
13) Write a C program to find out power of given number without using pow() function.	19
14) Write a C program to find the complement of a particular bit input at run time.	21
15) Write a C program to print palindrome numbers between 1 to 1000.	22
16) Write a C program to print Fibonacci series between 0 to 50.	23
17) Write a C program to find the given number is power of 2 or not.	24
18) Write a C program to find out the biggest number of three variables.	25
19) Accept a month in digit from the user. Display the month in words.	27
20) Write a C program to find the given number is Perfect number or not?	29

CONTROL STATEMENTS:-

1) Write a C Program to Find the factorial of a given number. User has to take the input at runtime.

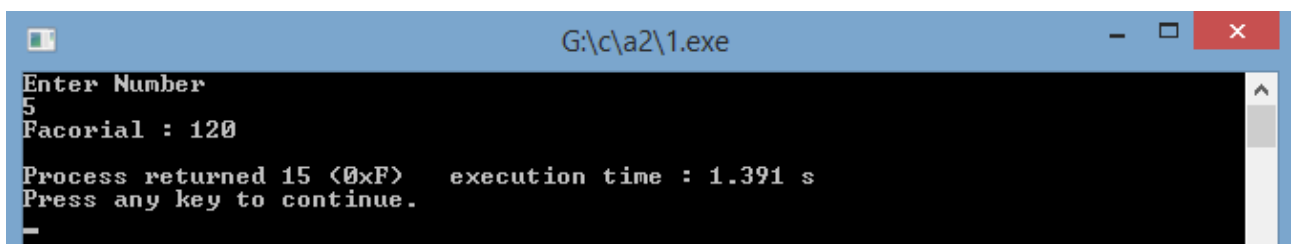
```
#include<stdio.h>

main()
{
    int fact=1,i,j,k,num;

    printf("Enter Number\n");
    scanf("%d",&num);

    for(i=1;i<=num;i++)
    {
        fact=fact*i;
    }

    printf("Facorial : %d\n",fact);
}
```



The screenshot shows a Windows command prompt window titled "G:\c\ a2\1.exe". The output of the program is as follows:

```
Enter Number
5
Facorial : 120
Process returned 15 (0xF)   execution time : 1.391 s
Press any key to continue.
-
```

2) Write a C Program to Find the sum of digits of a given number.

```
#include<stdio.h>

main()
{
    int s=0,num,i,j,k;

    printf("Enter Number\n");
    scanf("%d",&num);

    //for(s=0;num;s=s+num%10,num=num/10);

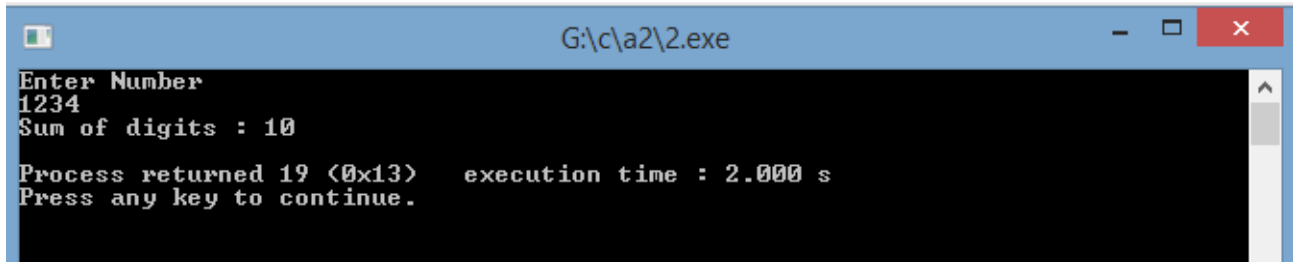
    /*
    while(num)
    {
        s=s+num%10;
        num=num/10;
    }
    */

    do
    {
        s = s + num % 10;
        num = num /10;
    }
```

```
while(num > 0);
```

```
printf("Sum of digits : %d\n",s);
```

```
}
```



```
Enter Number
1234
Sum of digits : 10

Process returned 19 (0x13)   execution time : 2.000 s
Press any key to continue.
```

3) Write a C Program to reverse the digits of a given number.

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

```
main()
```

```
{
```

```
    int s=0,i,j,k,num;
```

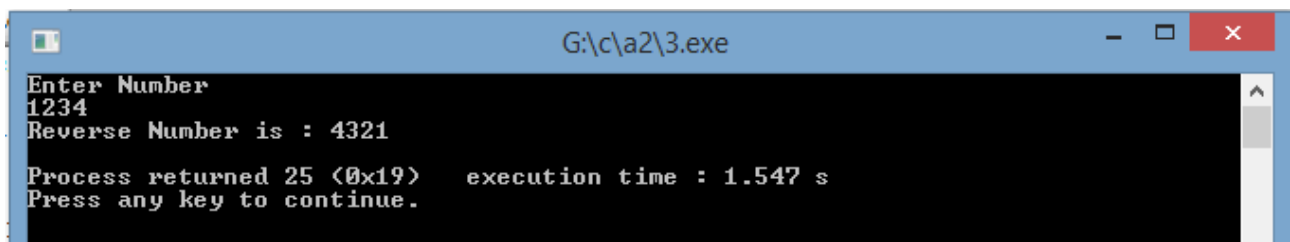
```
    printf("Enter Number\n");
```

```
    scanf("%d",&num);
```

```
    for(s=0;num;s=10*s+num%10,num=num/10);
```

```
    printf("Reverse Number is : %d\n",s);
```

```
}
```



The screenshot shows a Windows command prompt window titled "G:\c\a2\3.exe". The output of the program is as follows:

```
Enter Number
1234
Reverse Number is : 4321
Process returned 25 (0x19)    execution time : 1.547 s
Press any key to continue.
```

4) Write a C program to convert a character. If it is Lower, convert it to Upper and if it is Upper convert it to Lower character.

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

```
main()
```

```
{
```

```
    char c;
```

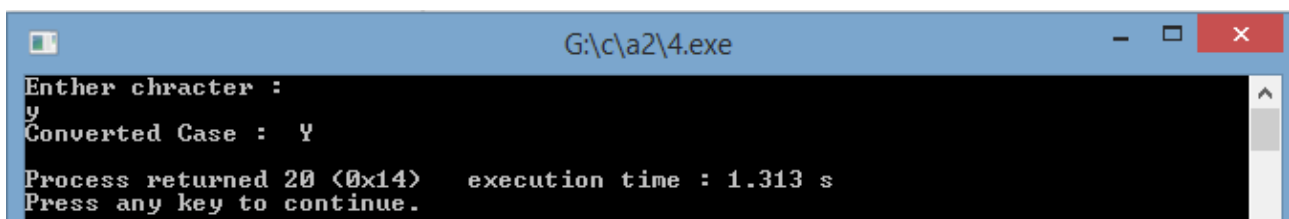
```
    printf("Enter chracter : \n");
```

```
    scanf("%c",&c);
```

```
    c = c ^ 32;
```

```
    printf("Converted Case : %c\n",c);
```

```
}
```



```
G:\c\a2\4.exe
Enter chracter :
y
Converted Case : Y
Process returned 20 (0x14)   execution time : 1.313 s
Press any key to continue.
```

5) Write a C program to print multiplication tables from 10 to 15.

```
#include<stdio.h>

main()
{
    int i,j;

    for(i=1;i<=10;i++)
    {
        for(j=11;j<=15;j++)
        {
            printf("%d*%2d=%3d| ",j,i*i*j);

        }
        printf("\n");
    }
}
```

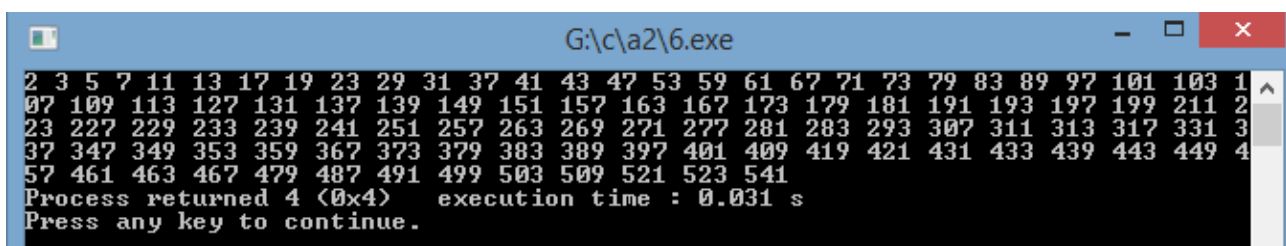
```
G:\c\a2\5.exe
11* 1= 11| 12* 1= 12| 13* 1= 13| 14* 1= 14| 15* 1= 15|
11* 2= 22| 12* 2= 24| 13* 2= 26| 14* 2= 28| 15* 2= 30|
11* 3= 33| 12* 3= 36| 13* 3= 39| 14* 3= 42| 15* 3= 45|
11* 4= 44| 12* 4= 48| 13* 4= 52| 14* 4= 56| 15* 4= 60|
11* 5= 55| 12* 5= 60| 13* 5= 65| 14* 5= 70| 15* 5= 75|
11* 6= 66| 12* 6= 72| 13* 6= 78| 14* 6= 84| 15* 6= 90|
11* 7= 77| 12* 7= 84| 13* 7= 91| 14* 7= 98| 15* 7=105|
11* 8= 88| 12* 8= 96| 13* 8=104| 14* 8=112| 15* 8=120|
11* 9= 99| 12* 9=108| 13* 9=117| 14* 9=126| 15* 9=135|
11*10=110| 12*10=120| 13*10=130| 14*10=140| 15*10=150|

Process returned 10 (0xA)   execution time : 0.031 s
Press any key to continue.
```

6) Write a C program to print first 100 prime numbers.

```
#include<stdio.h>

main()
{
    int num,i,j,k,l,c=0;
    for(j=1;c<100;j++)
    {
        for(i=2;i<j;i++)
        {
            if(j%i==0)
                break;
        }
        if(i==j)
        {
            c++;
            printf("%d ",j);
        }
    }
}
```



```
G:\c\al2\6.exe
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 211 223 227 229 233 239 241 251 257 263 269 271 277 281 283 293 307 311 313 317 331 337 347 349 353 359 367 373 379 383 389 397 401 409 419 421 431 433 439 443 449 457 461 463 467 479 487 491 499 503 509 521 523 541
Process returned 4 (0x4) execution time : 0.031 s
Press any key to continue.
```


7) Write a C program to print Armstrong Numbers between 1 to 500.

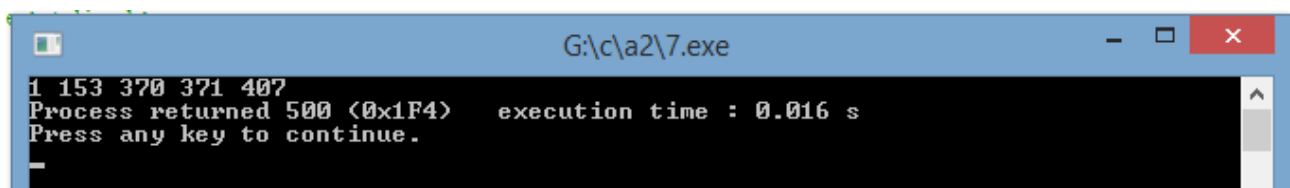
```
#include<stdio.h>

main()
{
    int d,i,j,k,num,s,m;

    for(j=1;j<=500;j++)
    {
        i=j;
        d=j;

        for(s=0;i;k=i% 10,s=s+(k*k*k),i=i/10);

        if(j==s)
            printf("%d ",j);
    }
}
```



```
G:\c\a2\7.exe
1 153 370 371 407
Process returned 500 (0x1F4)   execution time : 0.016 s
Press any key to continue.
```

8) Write a C program to print the binary of a given number (+ve or -ve numbers).

```
#include<stdio.h>

main()
{
    int num,i,c=0;

    printf("Enter number\n");

    scanf("%d",&num);

    for(i=sizeof(int)*8-1;i>=0;i--)
    {
        printf("%d",num>>i&1);

        if(num >> i&1 == 1)
            c++;

        if(i%8==0)
            printf(" ");

    }

    printf("\n1`s are = %d\n",c);

    printf("0`s are = %d\n",sizeof(int)*8-c);

    printf("\n");
}
```

9) Write a C program to reverse the bits of a given number.

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

```
main()
```

```
{
```

```
    int num,i,j;
```

```
    printf("Enter Number\n");
```

```
    scanf("%d",&num);
```

```
    for(i=sizeof(int)*8-1;i>=0;i--)
```

```
    {
```

```
        printf("%d",num >> i & 1);
```

```
        if (i % 8 == 0)
```

```
            printf(" ");
```

```
    }
```

```
    printf("\n");
```

```
    for(i=0,j=sizeof(int)*8-1;i<((sizeof(int)*8)/2;i++,j--)
```

```
    {
```

```
        if( (num >> i & 1) != (num >> j & 1) )
```

```
        {
```

```

        num = num ^ 1 << i;

        num = num ^ 1 << j;
    }

}

for(i=sizeof(int)*8-1;i>=0;i--)
{
    printf("%d",num >> i & 1);
    if (i % 8 == 0)
        printf(" ");
}

printf("\n");
}

```

```

G:\c\a2\9.exe
Enter Number
15
00000000 00000000 00000000 00001111
11110000 00000000 00000000 00000000

Process returned 10 (0xA)   execution time : 4.594 s
Press any key to continue.

```

10) Write a C program to implement a Calculator using switch.

```
#include<stdio.h>

main()
{
    int i,j;
    char c;

    printf("Enter Two Numbers : \n");
    scanf("%d %d",&i,&j);

    printf("Enter Choice : \n+ : Addition\n- : Subtraction\n* : Multiplication\n/ : Divison\n%% : Modulus\n");
    printf("Enter Character : \n");
    scanf(" %c",&c);

    switch(c)
    {
        case '+':
            printf("%d + %d = %d\n",i,j,i+j);
            break;

        case '-':
            printf("%d - %d = %d\n",i,j,i-j);
            break;
```

```
case '*':
```

```
printf("%d * %d = %d\n",i,j,i*j);
```

```
break;
```

```
case '/':
```

```
printf("%d / %d = %d\n",i,j,i/j);
```

```
break;
```

```
case '%':
```

```
printf("%d %% %d = %d\n",i,j,i%j);
```

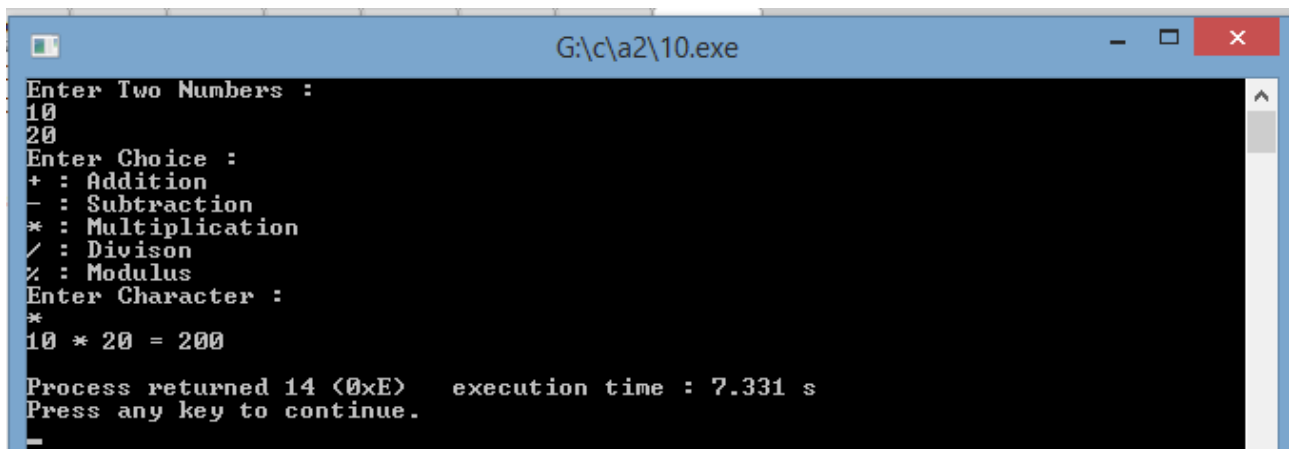
```
break;
```

```
default:
```

```
printf("Invalid\n");
```

```
}
```

```
}
```



```
G:\c\a2\10.exe
Enter Two Numbers :
10
20
Enter Choice :
+ : Addition
- : Subtraction
* : Multiplication
/ : Divison
% : Modulus
Enter Character :
*
10 * 20 = 200
Process returned 14 (0xE)   execution time : 7.331 s
Press any key to continue.
```

11) Write a C program to find the complement of a given number and then print it's binary decimal, octal and Hexa decimal values.

Note : Number can be either +ve or -ve. And observe the result satisfying its formulae or not.

Formulae : 1's complement of (x) = - (x+1)

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

```
main()
```

```
{
```

```
    int i,j,k,num;
```

```
    printf("Enter Number\n");
```

```
    scanf("%d",&num);
```

```
    num=~num;
```

```
    printf("\nDecimal : %d\n",num);
```

```
    for(i=sizeof(int)*8-1;i>=0;i--)
```

```
    {
```

```
        printf("%d",num>>i&1);
```

```
        if(i%8==0)
```

```
        printf(" ");  
    }  
  
    printf("\n");  
  
    printf("Octal : %o\n",num);  
    printf("Hex : %X\n",num);  
    printf("\n");  
  
}
```



```
G:\c\a2\11.exe  
Enter Number  
-98  
  
Decimal : 97  
00000000 00000000 00000000 01100001  
Octal : 141  
Hex : 61  
  
Process returned 10 (0xA)   execution time : 1.781 s  
Press any key to continue.  
_
```


12) Write a C program to find the 2's complement of a given number and then print its binary, decimal, octal and Hexa decimal values.

Note : Number can be either +ve or -ve and observe the result satisfying its Formula or not.

Formulae : 2's complement of (x) = - (x)

```
#include<stdio.h>

main()
{
    int i,j,k,num;

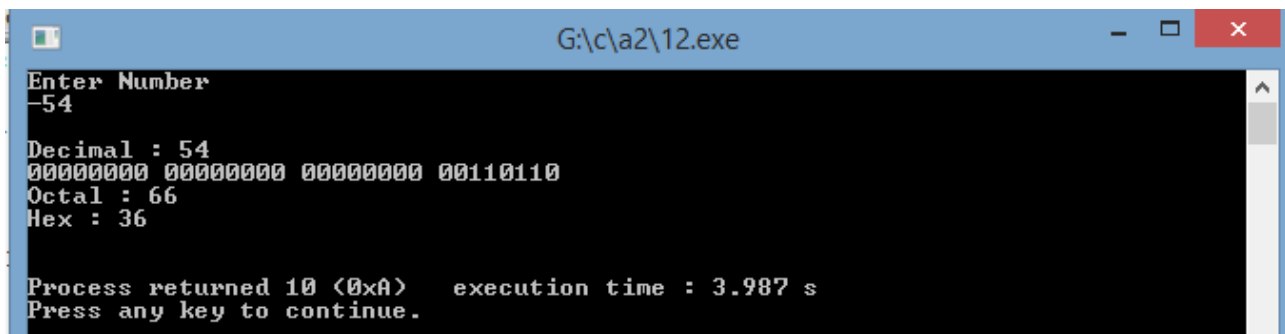
    printf("Enter Number\n");
    scanf("%d",&num);

    num=~num;
    num=num+1;

    printf("\nDecimal : %d\n",num);

    for(i=sizeof(int)*8-1;i>=0;i--)
    {
        printf("%d",num>>i&1);
        if(i%8==0)
```

```
        printf(" ");  
    }  
  
    printf("\n");  
  
    printf("Octal : %o\n",num);  
    printf("Hex : %X\n",num);  
    printf("\n");  
  
}
```



```
G:\c\a2\12.exe  
Enter Number  
-54  
Decimal : 54  
00000000 00000000 00000000 00110110  
Octal : 66  
Hex : 36  
  
Process returned 10 (0xA)   execution time : 3.987 s  
Press any key to continue.
```

13) Write a C program to find out power of given number without using pow() function.

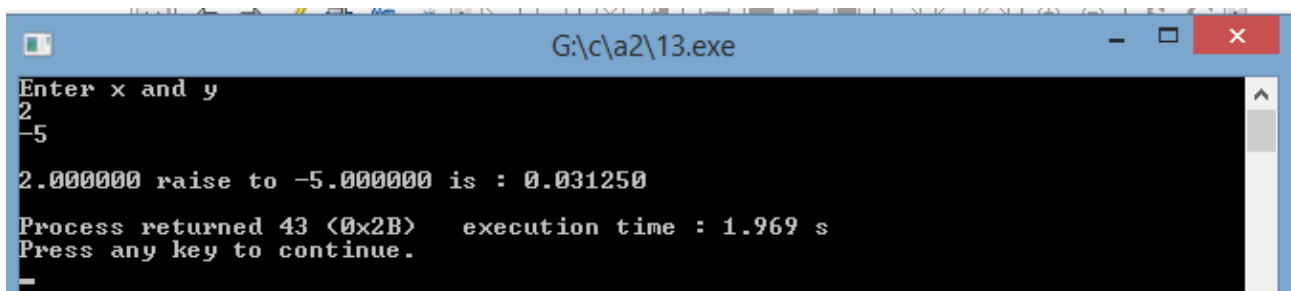
```
#include<stdio.h>

main()
{
    float m,x,y,i,pow=1,l,j,n;

    printf("Enter x and y\n");
    scanf("%f %f",&x,&y);

    if(y<0)
    {
        m=y;
        y=-y;
    }
    for(l=1;l<=y;l++)
    {
        pow=pow*x;
    }
    if(m<0)
        pow=(1/pow);

    printf("\n%f raise to %f is : %f\n",x,m,pow);
}
```



```
Enter x and y
2
-5
2.000000 raise to -5.000000 is : 0.031250
Process returned 43 (0x2B)   execution time : 1.969 s
Press any key to continue.
```

14) Write a C program to find the complement of a particular bit. User has to take the position number at runtime.

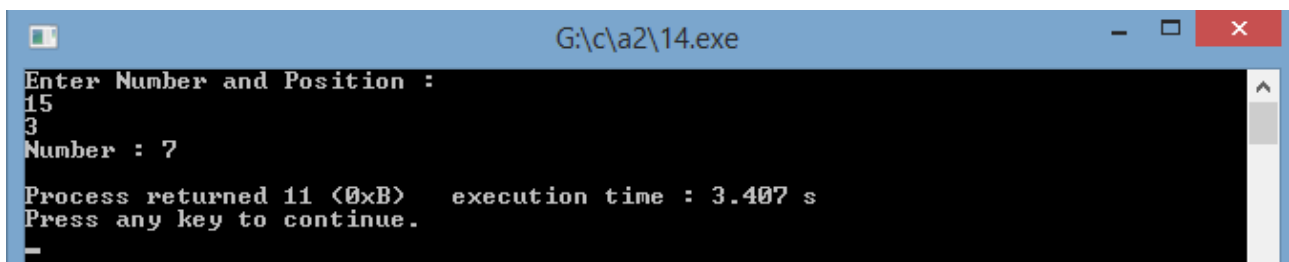
```
#include<stdio.h>

main()
{
    int num,pos;

    printf("Enter Number and Position : \n");
    scanf("%d %d",&num,&pos);

    num = num ^ 1 << pos;

    printf("Number : %d\n",num);
}
```



```
G:\c\a2\14.exe
Enter Number and Position :
15
3
Number : 7
Process returned 11 (0xB)   execution time : 3.407 s
Press any key to continue.
-
```

**15) Write a C program to print palindrome numbers between 1 to 1000.
(palindrome numbers means the numbers which are equal to its reverse.**

Ex : 11,22,33,44,.....999.)

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

```
main()
```

```
{
```

```
    int num,i,k,l,j,s=0,c=0;
```

```
    for(i=10;i<=1000;i++)
```

```
    {
```

```
        j=i;
```

```
        for(s=0;j;s=10*s+j%10,j=j/10);
```

```
        if(s == i)
```

```
        {
```

```
            printf("%d ",i);
```

```
            c++;
```

```
        }
```

```
    }
```

```
    printf("\nTotal:%d\n",c);
```

```
}
```

16) Write a C program to print Fibonacci series between 0 to 50.

```
#include<stdio.h>

main()
{
    int i,j=0,k=1,l=1,num;

    printf("%d %d ",j,k);

    for(i=0;l<=50;i++)
    {
        l=j+k,j=k,k=l;
        if(l<50)
            printf("%d ",l);

    }
    printf("\n");
}
```

17) Write a C program to find the given number is power of 2 or not.

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

```
main()
```

```
{
```

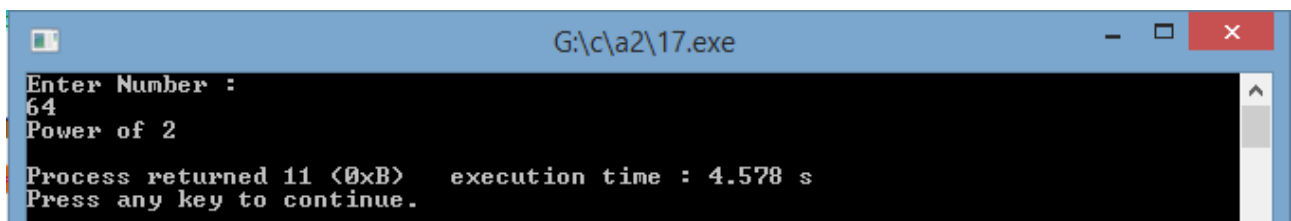
```
    int num;
```

```
    printf("Enter Number : \n");
```

```
    scanf("%d",&num);
```

```
    (num & num - 1) ? printf("Not..\n"):printf("Power of 2\n");
```

```
}
```

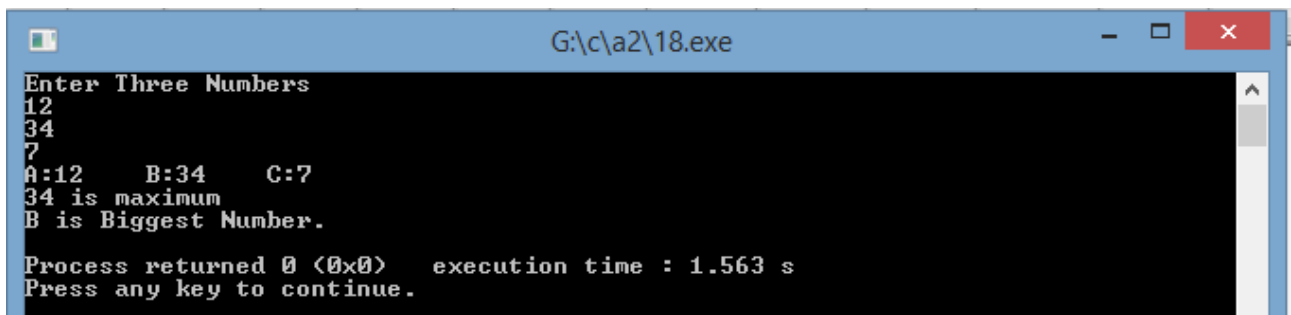


```
G:\c\2\17.exe
Enter Number :
64
Power of 2
Process returned 11 (0xB)   execution time : 4.578 s
Press any key to continue.
```


18) Write a C program to find out the biggest number of three variables using if-else Ladder and ternary operator.

```
#include<stdio.h>

main()
{
    unsigned int a,b,c,max;
    printf("Enter Three Numbers\n");
    scanf("%d %d %d",&a,&b,&c);
    printf("A:%u\tB:%u\tC:%u\n",a,b,c);
    max = (a>b)?(a>c) ? a : c : ((b>c)? b:c);
    printf("%d is maximum\n",max);
    if(a>b && a>c)
    {
        printf("A is Biggest Number.\n");
    }
    else if(b>a && b>c)
    {
        printf("B is Biggest Number.\n");
    }
    else
    {
        printf("C is Biggest Number.\n");
    }
}
```



```
Enter Three Numbers
12
34
7
A:12    B:34    C:7
34 is maximum
B is Biggest Number.

Process returned 0 (0x0)   execution time : 1.563 s
Press any key to continue.
```

19) Accept a month in digit from the user. Display the month in words. If number is not between 1 and 12 display message “Invalid Month”. (Use ‘switch’)

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

```
main()
```

```
{
```

```
    int month,choice;
```

```
    printf("Enter the Month (1 to 12): \n");
```

```
    scanf("%d",&month);
```

```
    switch(month)
```

```
    {
```

```
        case 1: printf("January\n");
```

```
            break;
```

```
        case 2: printf("February\n");
```

```
            break;
```

```
        case 3: printf("March\n");
```

```
            break;
```

```
        case 4: printf("April\n");
```

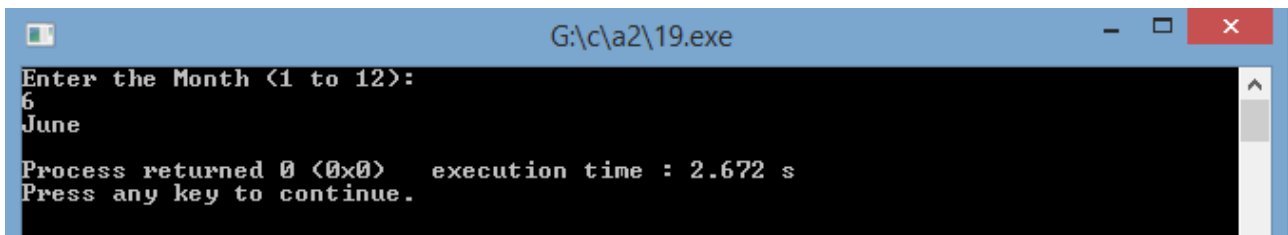
```
            break;
```

```
        case 5: printf("May\n");
```

```
            break;
```

```
        case 6: printf("June\n");
```

```
        break;
    case 7: printf("July\n");
        break;
    case 8: printf("August\n");
        break;
    case 9: printf("September\n");
        break;
    case 10: printf("October\n");
        break;
    case 11: printf("November\n");
        break;
    case 12: printf("December\n");
        break;
    default: printf("Invalid Month\n");
        break;
}
}
```



```
G:\c\a2\19.exe
Enter the Month <1 to 12>:
6
June
Process returned 0 (0x0)   execution time : 2.672 s
Press any key to continue.
```

20) Write a C program to find the given number is Perfect number or not?

Note : Perfect number means sum of it's divisors except that num is equalent to the Same number.

Ex : i/p num = 6.

6 diviesers are = 1,2,3, & 6.

sum = 1+2+3

sum = 6. So here 6 is perfect number.

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

```
main()
```

```
{
```

```
    int l,i,j,k,num,s=0;
```

```
    for(j=1;j<=10000;j++)
```

```
    {
```

```
        k=j;
```

```
        l=j;
```

```
        for(i=1;i<k;i++)
```

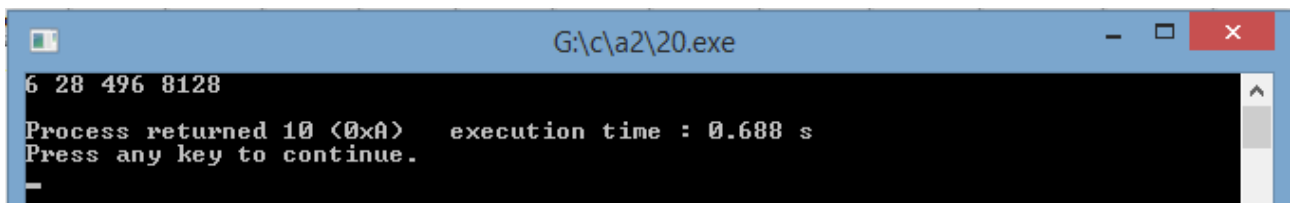
```
        {
```

```
            if(k%i==0)
```

```
            s=s+i;
```

```
        }
```

```
    if(s==k)
        printf("%d ",j);
    s=0;
}
printf("\n");
}
```



```
G:\c\a2\20.exe
6 28 496 8128
Process returned 10 (0xA) execution time : 0.688 s
Press any key to continue.
```