1. **HCF**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int a, b, small, i, hcf;

a=atoi(argv[1]);

b=atoi(argv[2]);

if(a<b)

small=a;

else

small=b;

for(i=1; i<=small; i++)

{

if((a%i==0)&&(b%i==0))

hcf=i;

}

printf("%d", hcf);

return 0;

}

1. **LCM**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int a, b, small, i, hcf, lcm;

a=atoi(argv[1]);

b=atoi(argv[2]);

if(a<b)

small=a;

else

small=b;

for(i=1; i<=small; i++)

{

if((a%i==0)&&(b%i==0))

hcf=i;

}

lcm=(a\*b)/hcf;

printf("%d", lcm);

return 0;

}

1. **FACTORIAL**

#include <stdio.h>

int main(int argc, char \*argv[])

{

int n, fact=1, i;

n=atoi(argv[1]);

for(i=1; i<=n; i++)

fact=fact\*i;

printf("%d", fact);

return 0;

}

1. **AREA OF CIRCLE**

#include <stdio.h>

#define PI 3.14

int main(int argc, char \*argv[])

{

int dia, rad, area;

dia=atoi(argv[1]);

rad=dia/2;

area=PI\*rad\*rad;

printf("%d", area);

return 0;

}

1. **LEAP YEAR**

#include <stdio.h>

int main(int argc, char \*argv[])

{

int year;

year=atoi(argv[1]);

if(year%100==0)

{

if(year%400==0)

printf("Leap Year");

else

printf("Not a Leap Year");

}

else if(year%4==0)

printf("Leap Year");

else

printf("Not a Leap Year");

return 0;

}

1. **AREA OF TRIANGLE**

#include <stdio.h>

int main(int argc, char \*argv[])

{

float area, base, height;

base=atoi(argv[1]);

height=atoi(argv[2]);

area=0.5\*base\*height;

printf("%.2f", area);

return 0;

}

1. **FIBONACCI SERIES**

#include <stdio.h>

int main(int argc, char \*argv[])

{

int i, n, first=0, second=1,next;

n=atoi(argv[1]);

for(i=0; i<n; i++)

{

printf("%d ", first);

next=first+second;

first=second;

second=next;

}

return 0;

}

1. **HYPOTENUSE**

#include<stdio.h>

#include<math.h>

int main(int argc, char \*argv[])

{

float a, b, h;

a=atoi(argv[1]);

b=atoi(argv[2]);

h=sqrt((a\*a)+(b\*b));

printf("%.2f", h);

return 0;

}

1. **PRIME NUMBER**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, count, i;

num=atoi(argv[1]);

for(i=1; i<=num; i++)

{

if(num%i==0)

count++;

}

if(count==2)

printf("Prime Number");

else

printf("Not a Prime Number");

return 0;

}

1. **. DECIMAL TO BINARY**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, i=0, j, binarynum[30];

num=atoi(argv[1]);

while(num)

{

binarynum[i]=num%2;

num=num/2;

i++;

}

for(j=i-1; j>=0; j--)

printf("%d", binarynum[j]);

return 0;

}

1. **. CHECK PALINDROME**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, num1, rev=0, rem;

num=atoi(argv[1]);

num1=num;

while(num1>0)

{

rem=num1%10;

rev=rev\*10+rem;

num1=num1/10;

}

if(num==rev)

printf("Palindrome");

else

printf("Not a Palindrome");

return 0;

}

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, num1, rev=0, rem, l=0, s=9;

num=atoi(argv[1]);

num1=num;

while(num1>0)

{

rem=num1%10;

if(rem<s)

s=rem;

if(rem>l)

l=rem;

rev=rev\*10+rem;

num1=num1/10;

}

if(num==rev)

printf("%d", l);

else

printf("%d", s);

return 0;

}

**12. PERFECT SQUARE**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, i;

num=atoi(argv[1]);

for(i=1; i<=num; i++)

{

if(num==(i\*i))

{

printf("Yes");

return 0;

}

}

printf("No");

return 0;

}

**13. STRONG NUMBER**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, count, fact, rem, sum=0, temp;

num=atoi(argv[1]);

temp = num;

while(num>0)

{

count = 1, fact = 1;

rem = num % 10;

while(count <= rem)

{

fact = fact \* count;

count++;

}

sum = sum + fact;

num = num / 10;

}

if(sum==temp)

printf("Strong Integer");

else

printf("Not a Strong Integer");

return 0;

}

**14. ARMSTRONG NUMBER**

#include<stdio.h>

int main(int argc, char \*argv[])

{

int num, sum=0, rem=0, cube=0, temp;

num=atoi(argv[1]);

temp=num;

while(num!=0)

{

rem=num%10;

cube=rem\*rem\*rem;

sum=sum+cube;

num=num/10;

}

if(sum==temp)

printf("Armstrong");

else

printf("No");

return 0;

}