***NAME : Himanshu Dixit***

***ENROLL NO. : B64178***

***BATCH : B10***

***Software Development Fundamentals – I(15B11CI111)***

***ODD 2021***

***Tutorial Sheet – 8***

***Q1. [CO4]****Answer the following (a) and (b) both.*

***(a)*** *What are actual and formal parameters in call by value functions?*

***Solution:***

|  |  |
| --- | --- |
| **Actual parameters** | ***Formal parameters*** |
| **1.** used in function calling.  **2.** are the variables constant or expressions contained in a function call that replace the formal parameters which is a part of the function declaration/definition. | **1.** used in function declaration.  **2.** also called dummy arguments.  **3.** may also replaced by the actual parameters.  **4.** may be declared by the same name or different name as of actual parameters. |

***(b)*** *What are the features of call by value functions?*

***Solution:***

The main feature of call by value that formal argument not affect the value of actual arguments.

***Q2. [CO4]****Write a program in C to find the sum of the series 1!/1+2!/2+3!/3+4!/4+5!/5 using the function.*

***Solution:***

#include<stdio.h>

int fact(int f);

int main()

{

float sum=0;

int n,a;

scanf("%d",&n);

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

{

printf("%d!/%d",i,i);

if(i!=n)

printf(" + ");

a=fact(i);

sum=sum+(a/i);

}

printf("\nsum = %.2f",sum);

return 0;

}

int fact(int f)

{

int i=1,p=1;

while(i<=f)

{

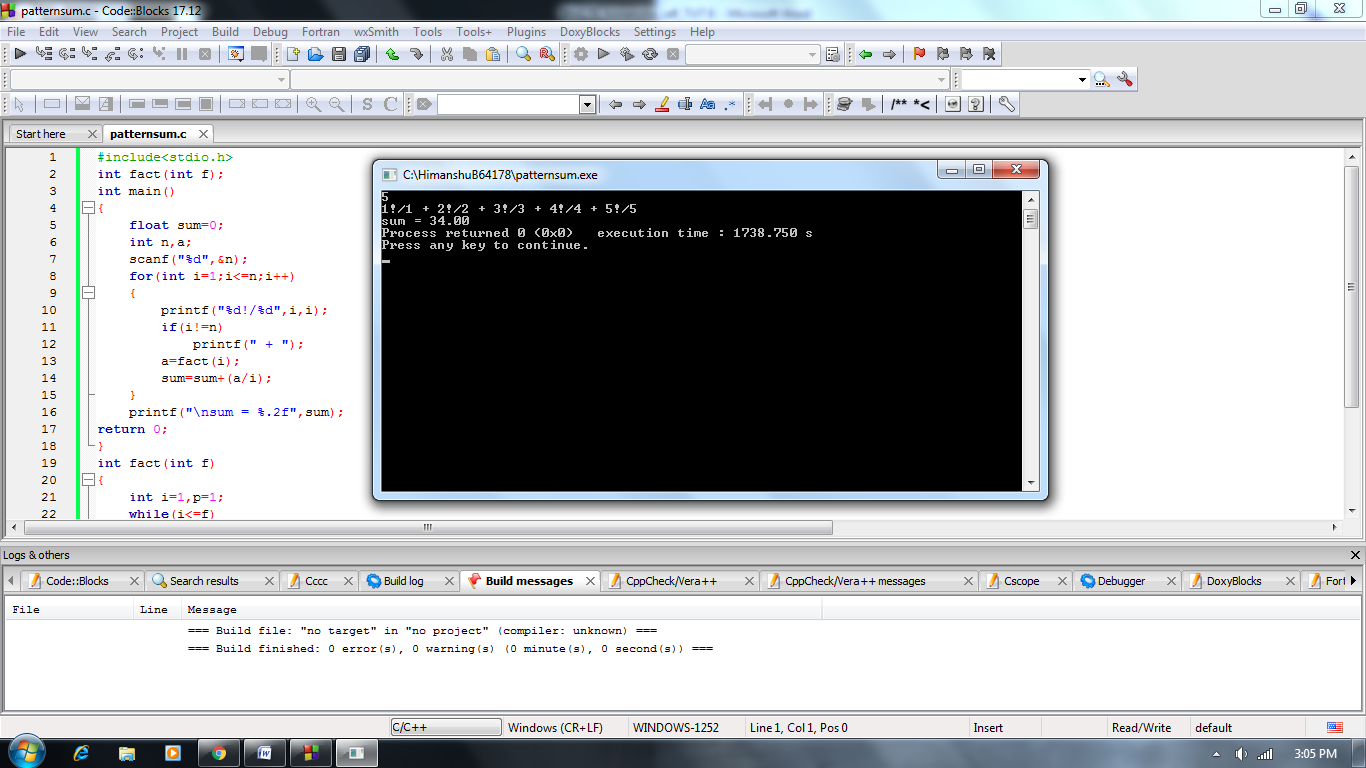
p=p\*i;

i++;

}

return p;

}



***Q3. [CO4]****What will be the output of the following program?*

*#include<stdio.h>*

*void modify (int n) {*

*n=n+10;*

*printf("Value of ‘n’ inside function=%d \n", n);*

*}*

*int main() {*

*int x=15;*

*modify(x);//passing value in function*

*printf("Modified value of x=%d \n", x);*

*return 0;*

*}*

***Solution:***

Value of ‘n’ inside function=25

Modified value of x=15

***Q4. [CO4]****A girl carries 50 hair pins in a haphazard manner within her dressing drawer. Her mother asked her to arrange these pins so that the drawer will look elegant. But the girl is confused as she does not know the number of possible ways in which she can arrange her hair pins. You must suggest a recursive function to the girl in order to ease her task.*

***Solution:***

#include<stdio.h>

unsigned long long int recfact(int fact)

{

if(fact==1)

return 1;

else

{

return((unsigned long long int)(fact\*recfact(fact-1)));

}

}

int main()

{

unsigned long long int f;

int nop=50;

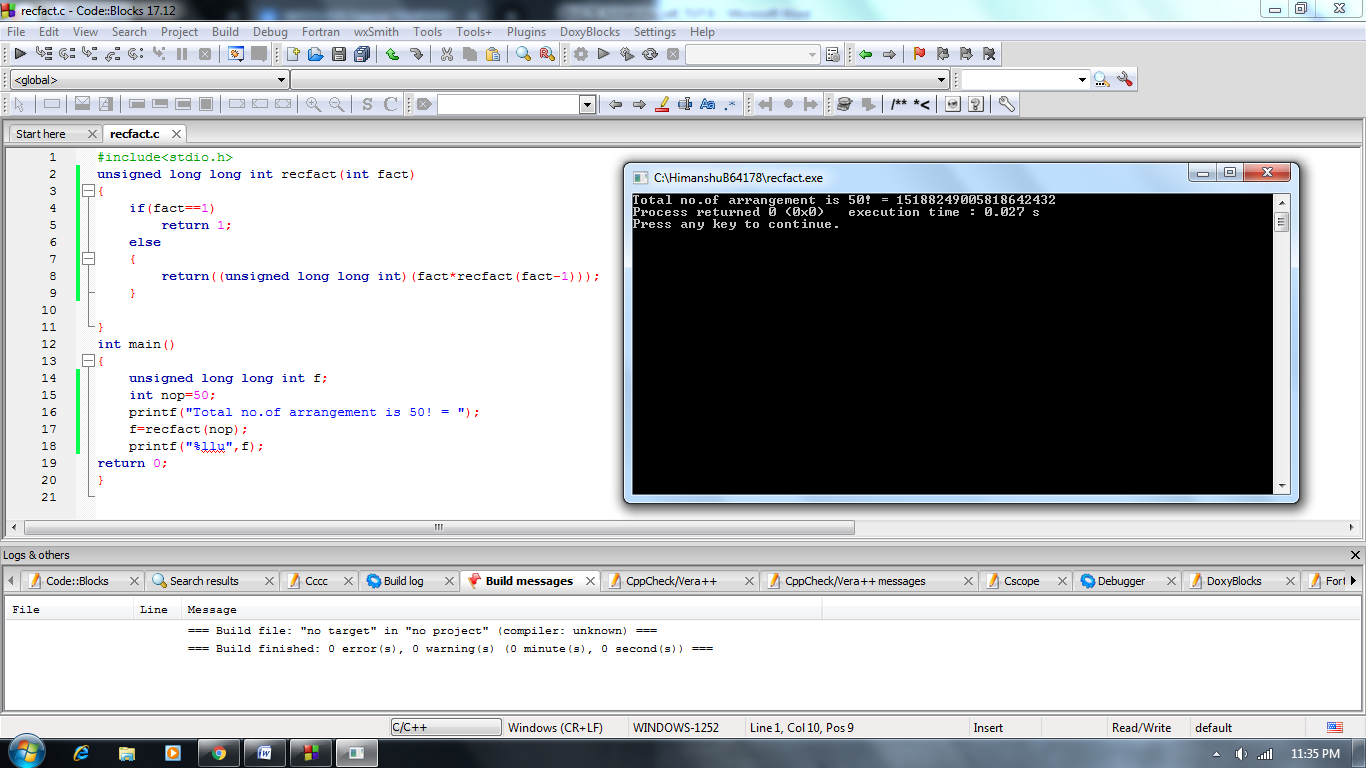
printf("Total no.of arrangement is 50! = ");

f=recfact(nop);

printf("%llu",f);

return 0;

}



***Q5. [CO4]****Write a program in C to convert decimal number to binary number using the function.*

***Solution:***

#include<stdio.h>

void conv(int n)

{

int i;

int a[10];

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

{

a[i]=n%2;

n=n/2;

}

int k=i-1;

printf("Binary convert is ");

for(i=k;i>=0;i--)

printf("%d",a[i]);

}

int main()

{

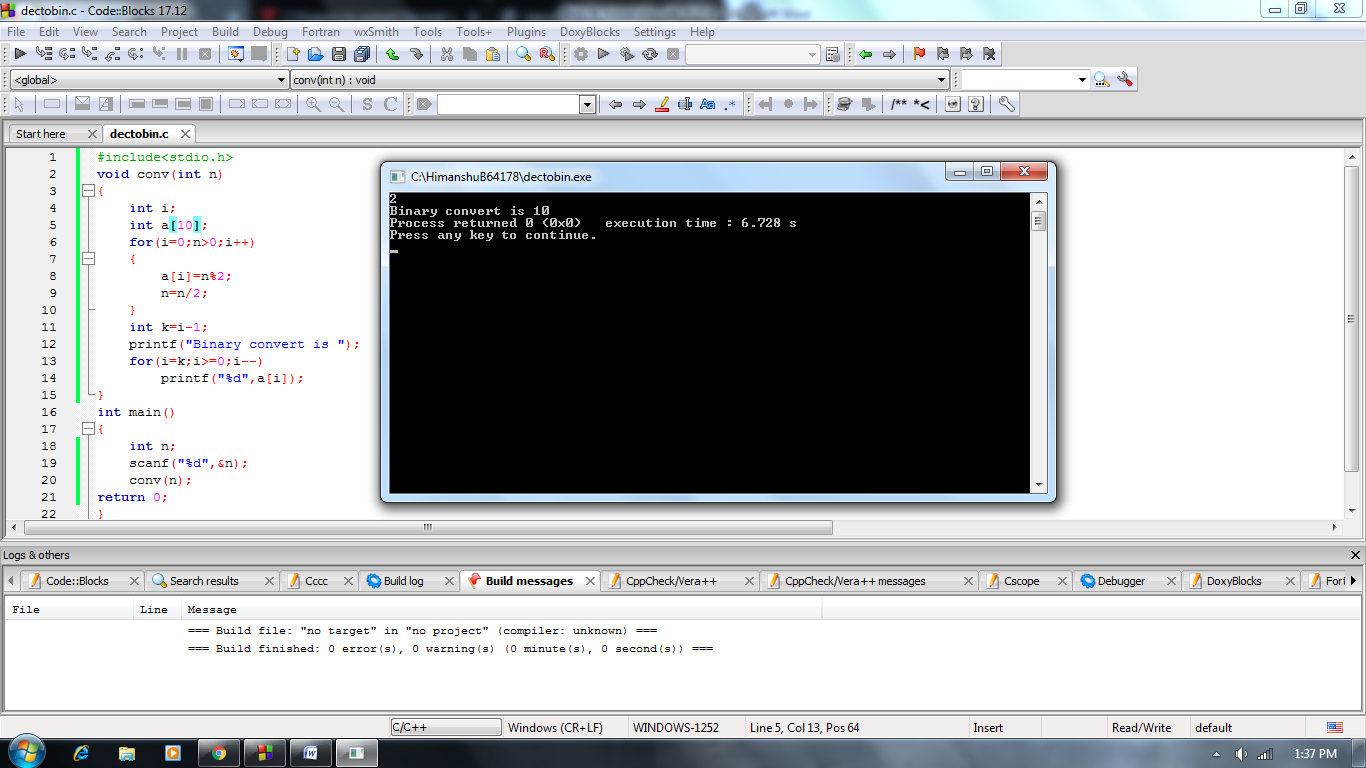
int n;

scanf("%d",&n);

conv(n);

return 0;

}



***Q6. [CO4]****Write a program to swap two numbers temporarily using call by value.*

***Solution:***

#include<stdio.h>

void swaping(int a,int b)

{

int k;

k=b;

b=a;

a=k;

printf("a = %d\nb = %d",a,b);

}

int main()

{

int a,b;

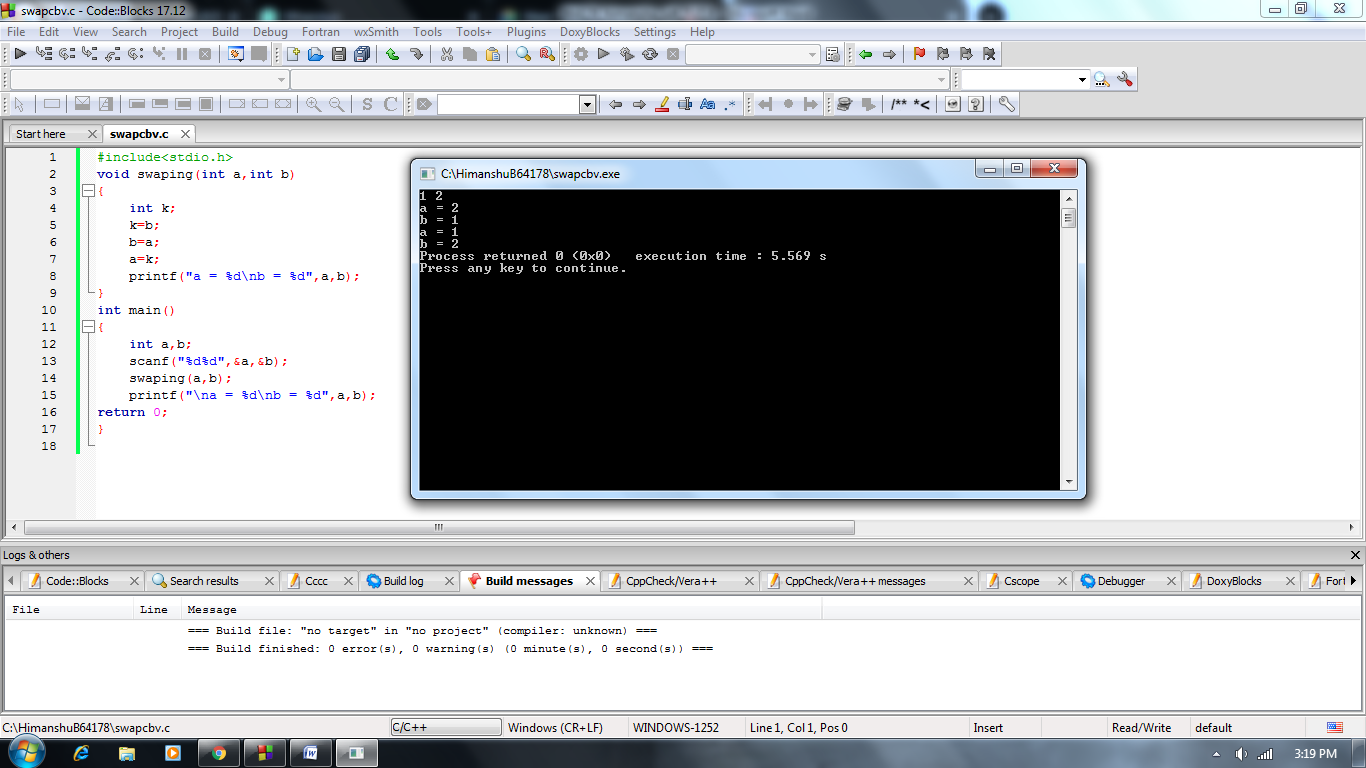
scanf("%d%d",&a,&b);

swaping(a,b);

printf("\na = %d\nb = %d",a,b);

return 0;

}



***Q7. [CO4]****A child is taking his daily lessons on mathematics from online classes. Today he is learning about the counting of stars. He starts counting as one star, two stars, three stars … then 49 stars and finally 50 stars. Help the child in summing up his counting of the number of stars using recursive function.*

***Solution:***

#include<stdio.h>

int recsum(int n,int sum)

{

if(n==51)

return sum;

sum=sum+n;

n++;

int s=recsum(n,sum);

return s;

}

int main()

{

int n=1,sum=0;

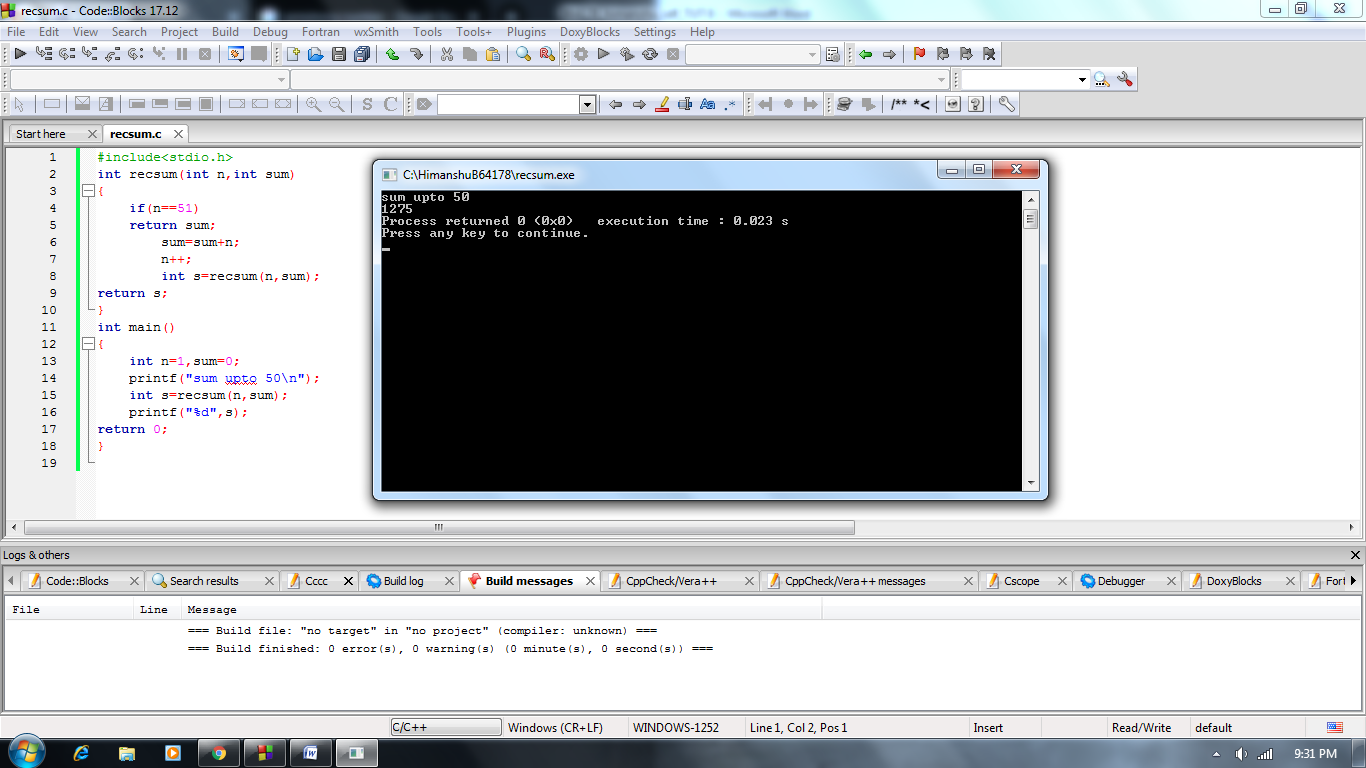
printf("sum upto 50\n");

int s=recsum(n,sum);

printf("%d",s);

return 0;

}



***Q8. [CO4]****Write a program in C to check armstrong and perfect numbers using the function.*

***Solution:***

#include<stdio.h>

void armstrong(int a)

{

int n,num=a,sum=0,count=0;

while(a>0)

{

a=a/10;

count++;

}

int i=count;

a=num;

while(a>0)

{

int k=1;

n=a%10;

a=a/10;

while(count>0)

{

k=k\*n;

count--;

}

sum=sum+k;

count=i;

}

if(num==sum)

printf("%d is armstrong number\n",num);

else

printf("%d is not armstrong number\n",num);

}

void perfect(int a)

{

int i=1,sum=0,num=a;

while(i<=a)

{

if(a%i==0)

sum=sum+i;

i++;

}

if(num\*2==sum)

printf("%d is perfect number\n",num);

else

printf("%d is not perfect number\n",num);

}

int main()

{

int a;

scanf("%d",&a);

armstrong(a);

perfect(a);

return 0;

}

