Tut 8

NBTG13715

Nitin Chaudhary

F8

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

1. What are actual and formal parameters in call by value functions?

Ans- Actual parameters -The parameters that appear in function calls.

Ans- Formal parameters -The parameters that appear in function declarations

Example-

#include<stdio.h>

int f(int a)//a is formal parameter

{

printf("%d",a\*2);

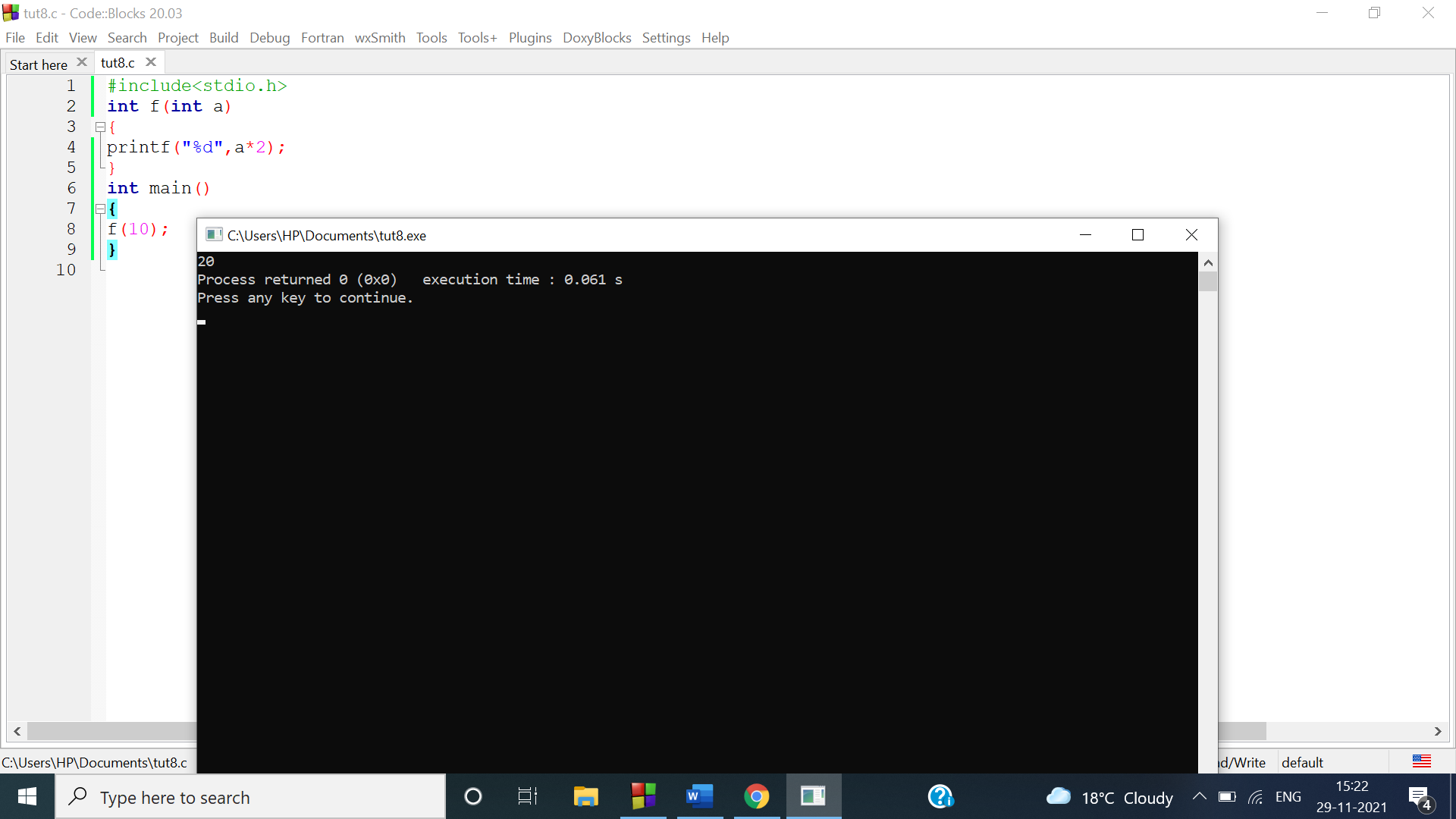
}

int main()

{

f(10);//10 is actual parameter

}



1. What are the features of call by value functions?

Ans-

The call by value method of passing arguments to a **function copies the actual value of an argument into the formal parameter of the function**. In this case, changes made to the parameter inside the function have no effect on the argument

Q2

#include<stdio.h>

int fact(int n)

{

int i,f=1;

for(n;n>0;n--)

f\*=n;

}

int main()

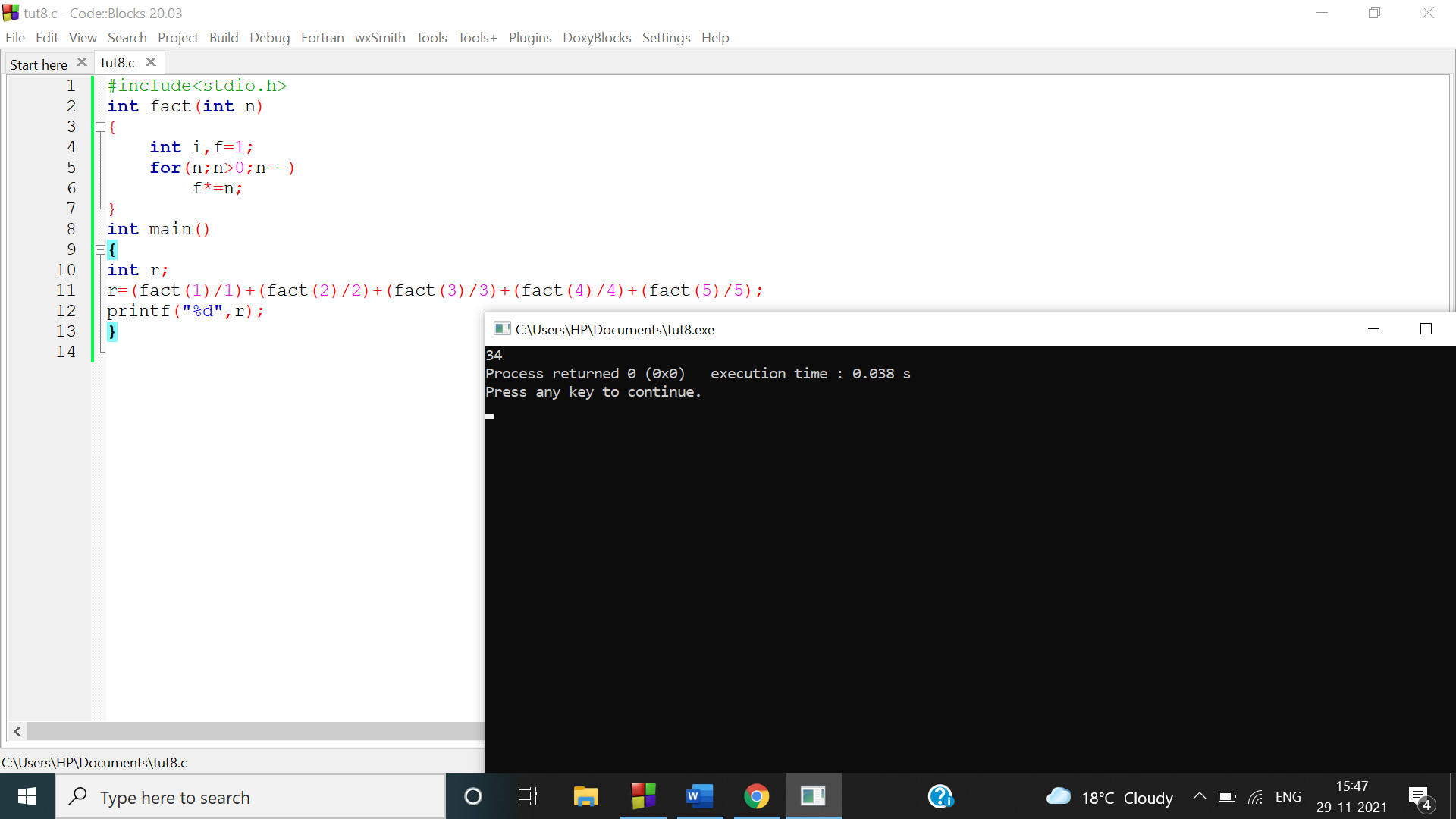
{

int r;

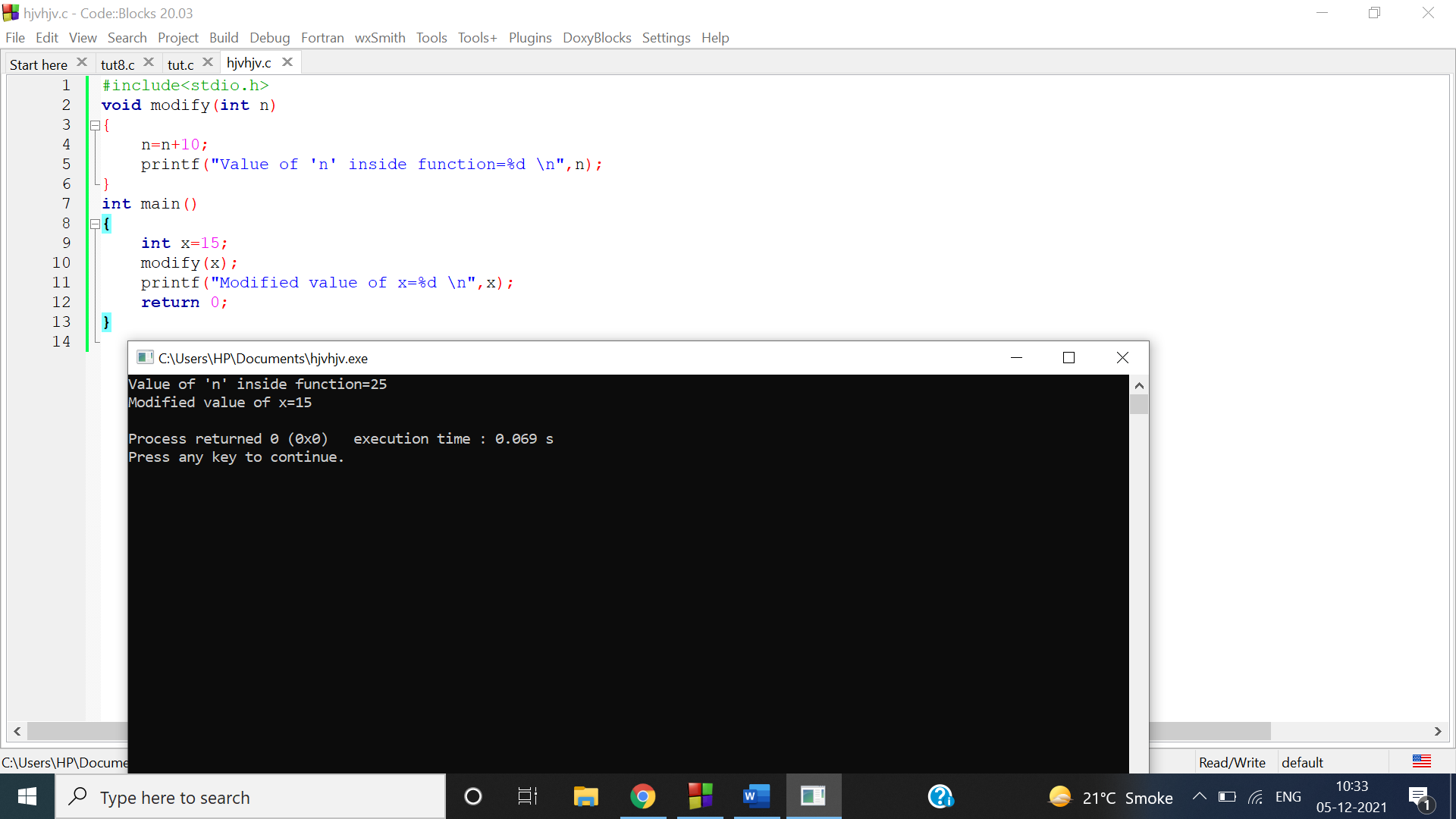
r=(fact(1)/1)+(fact(2)/2)+(fact(3)/3)+(fact(4)/4)+(fact(5)/5);

printf("%d",r);

}



Q3



This output is due to execution of functions

Q4

#include <stdio.h>

int fact(long long int n)

{

if (n==1)

{

return 1;

}

else

{

return n\*fact(n-1);

}

}

int main()

{

printf("Enter number of pins. ");

int n;

scanf("%d",&n);

long long int s=fact(n);

printf("The number of ways to arrange %d pins are %lld",n,s);

return 0;

}

Q5