StrongP

Software Documentation

Author: matiwa

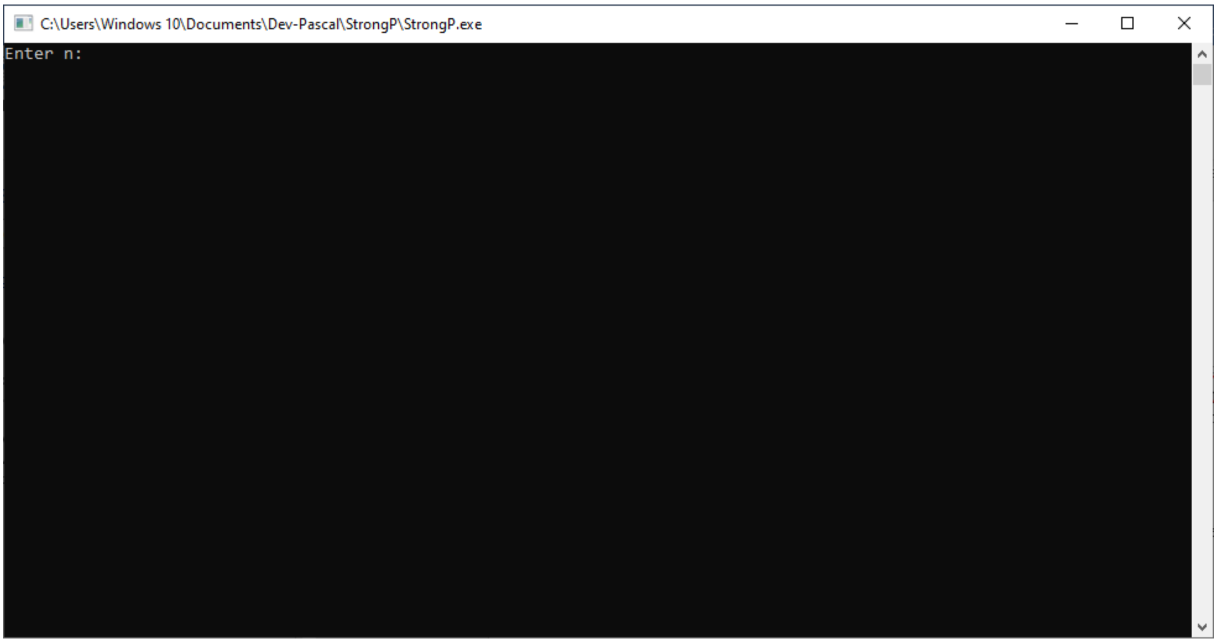
Table of contents

Introduction

This software documentation includes: description of the application's operation, what is needed for use, algorithms used, interface description and source code description. The application is used to calculate the strong of a selected non-negative integer n.

Describing of the application’s operation

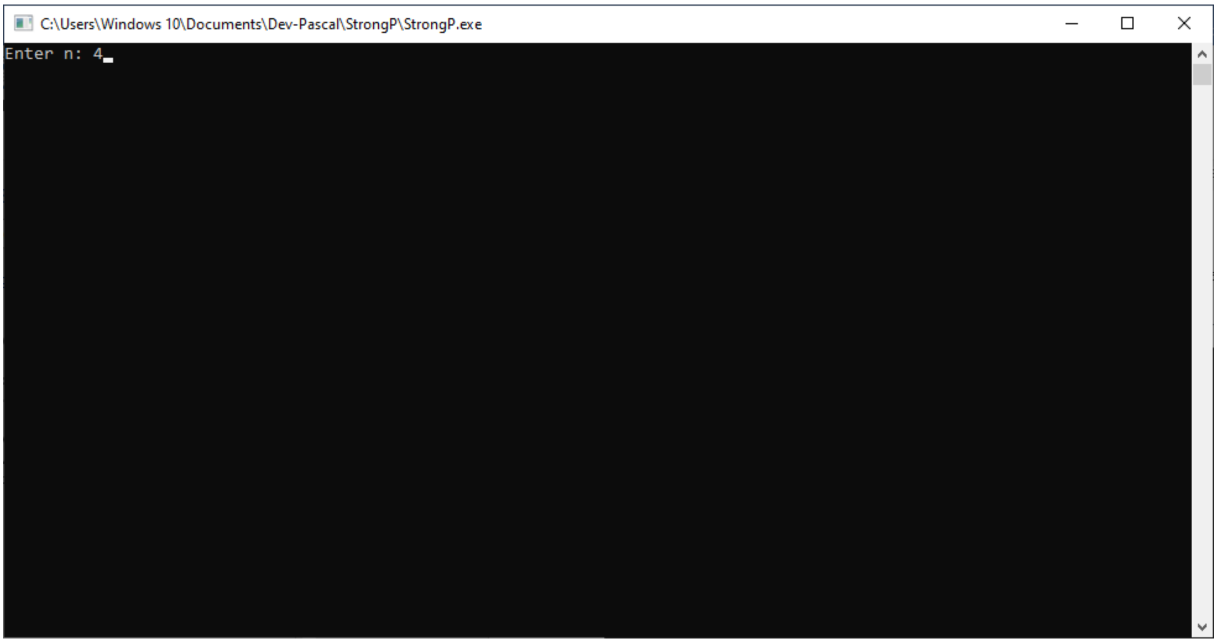
If user wants to calculate the strong, run the *.exe file. After this operation, the console window will appear on the screen as below.



```
C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe
Enter n:
```

Drawing 1: The beginning of the application's operation [own study]

User must enter a number that is non-negative and integer. Take the number 4 as an example.

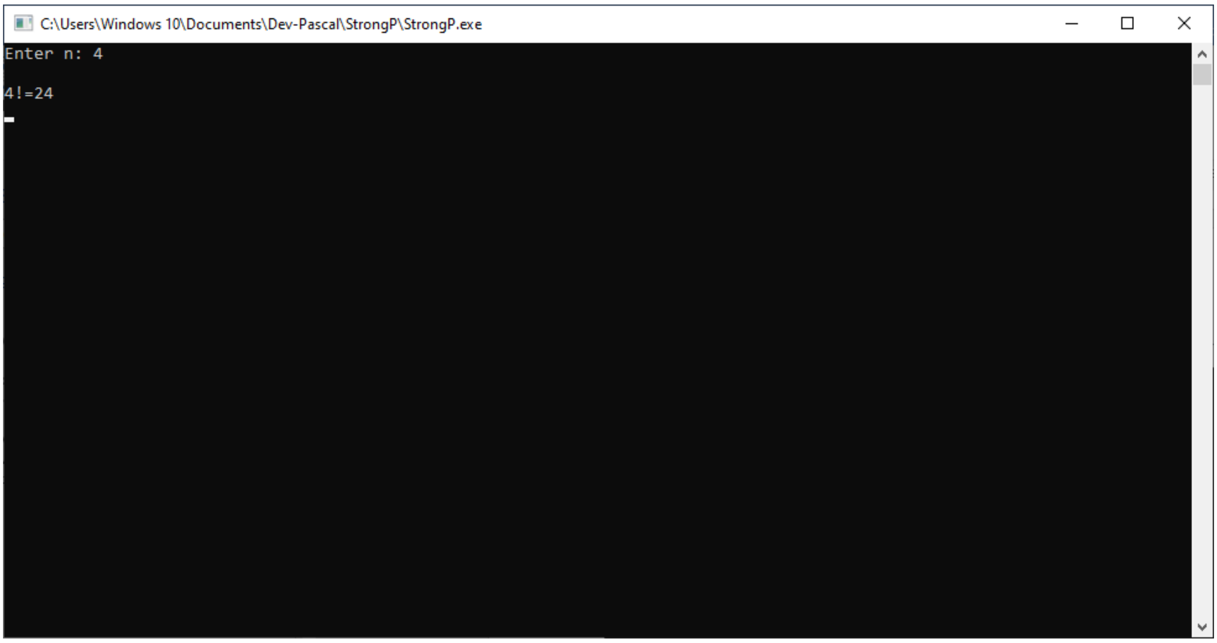


C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

```
Enter n: 4_
```

Drawing 2: The user enters a numer [own study]

After entering the number, the application will calculate the strong.



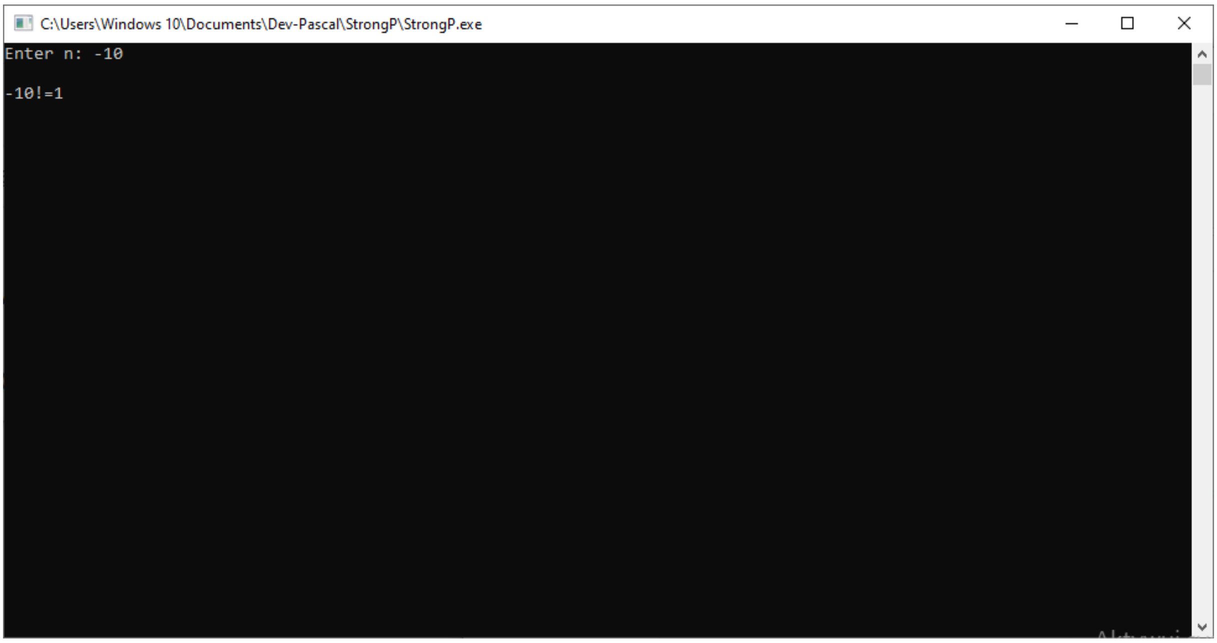
C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

```
Enter n: 4

4!=24
_
```

Drawing 3: The final result [own study]

If user clicks the any button, the application will end. This is the correct use of the program. There may be times when the user makes a mistake. The input number should be non-negative and integer. If user supplies a negative and an integer, the application will output 1.



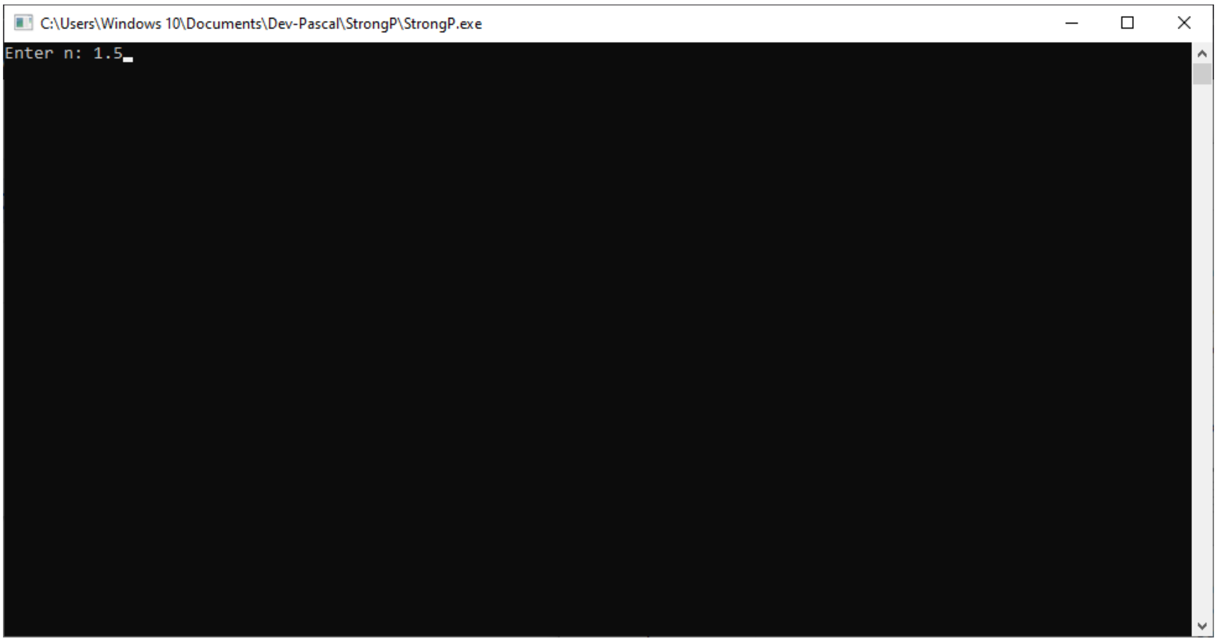
C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

```
Enter n: -10

-10!=1
```

Drawing 4: First mistake [own study]

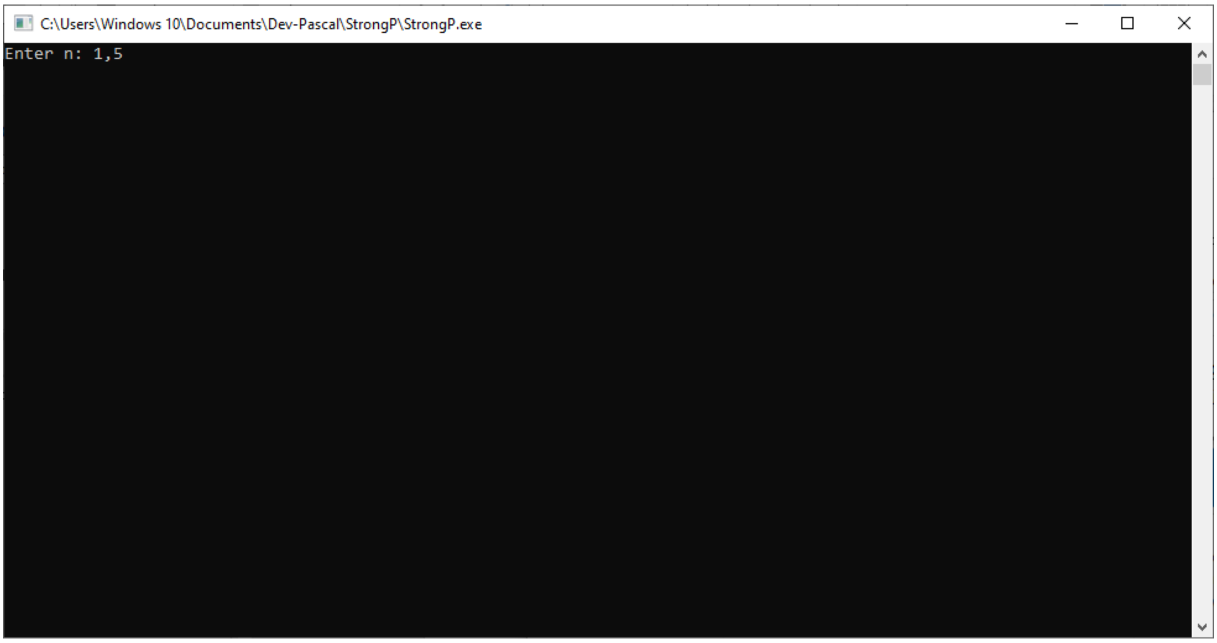
The second mistake is to give a floating point number, which math defines as rational.



C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

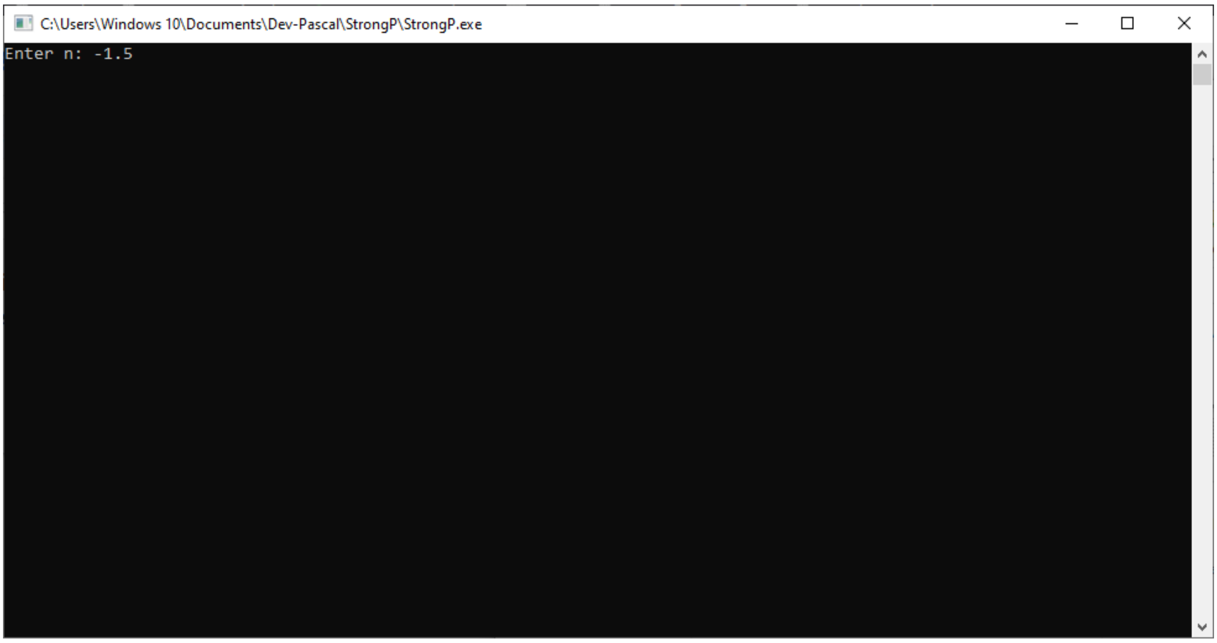
```
Enter n: 1.5_
```

Drawing 5: Second mistake [own study]


C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

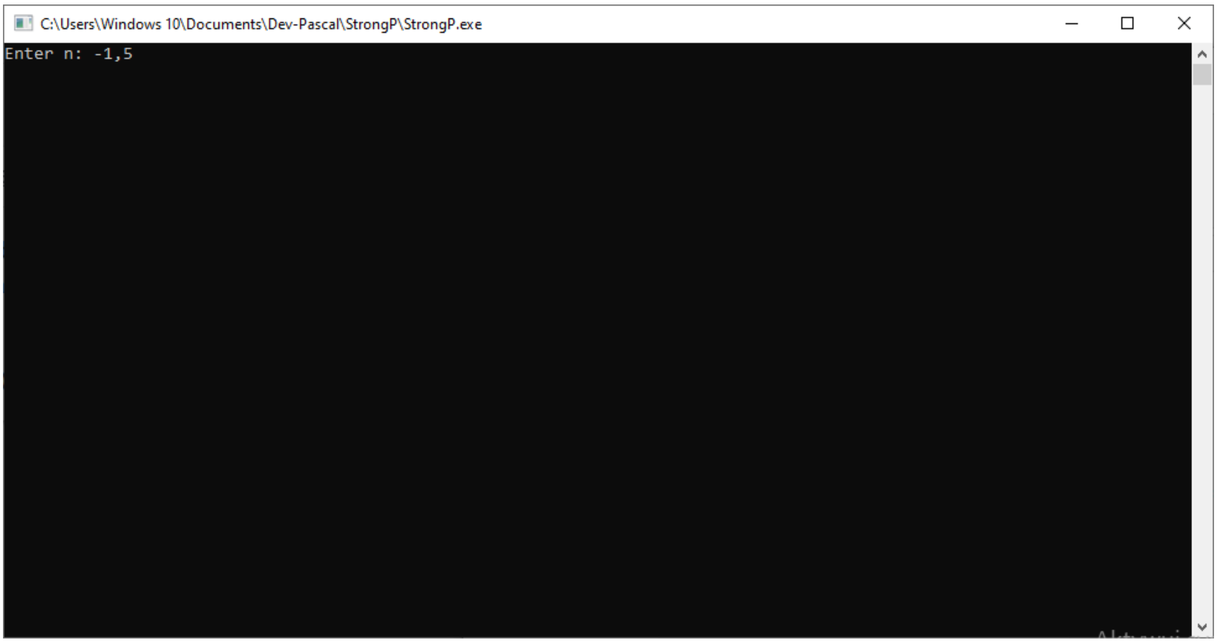
```
Enter n: 1,5
```

Drawing 6: Third mistake [own study]


C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

```
Enter n: -1.5
```

Drawing 7: Fourth mistake [own study]



C:\Users\Windows 10\Documents\Dev-Pascal\StrongP\StrongP.exe

```
Enter n: -1,5
```

Drawing 8: Fifth mistake [own study]

There are four cases of this type of error listed above. It doesn't matter if the user enters a comma or a period, the application will exit without any additional message. There are four cases of this type of error listed above. It doesn't matter if the user enters a comma or a period, the application will exit without any additional message. If he makes a mistake and enters a negative number, the already mentioned value 1 will be displayed and the program will wait for pressing any key.

What is needed for use?

The application does not require installation. It only needs the Windows operating system.

Algorithms used

The only algorithm used in application development is to calculate the strong of any non-negative and integer number. For n equal to 0, the strong is 1. If n is greater than or equal to 1, the strong for 0 will be 1 multiplied by the series of numbers preceding n and then itself. An example for the numbers n of 4, 5, and 6 is below.

n = 4

n! = 1 * 2 * 3 * 4   4! = 24

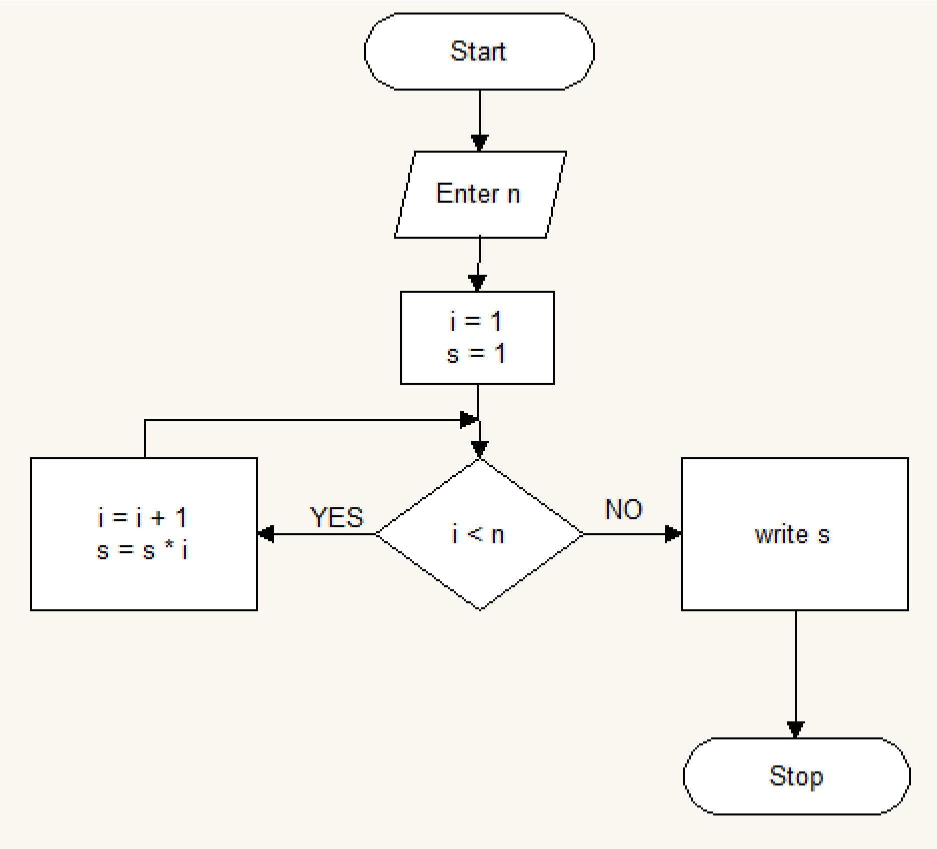
n = 5

n! = 1 * 2 * 3 * 4 * 5   5! = 120

n = 6

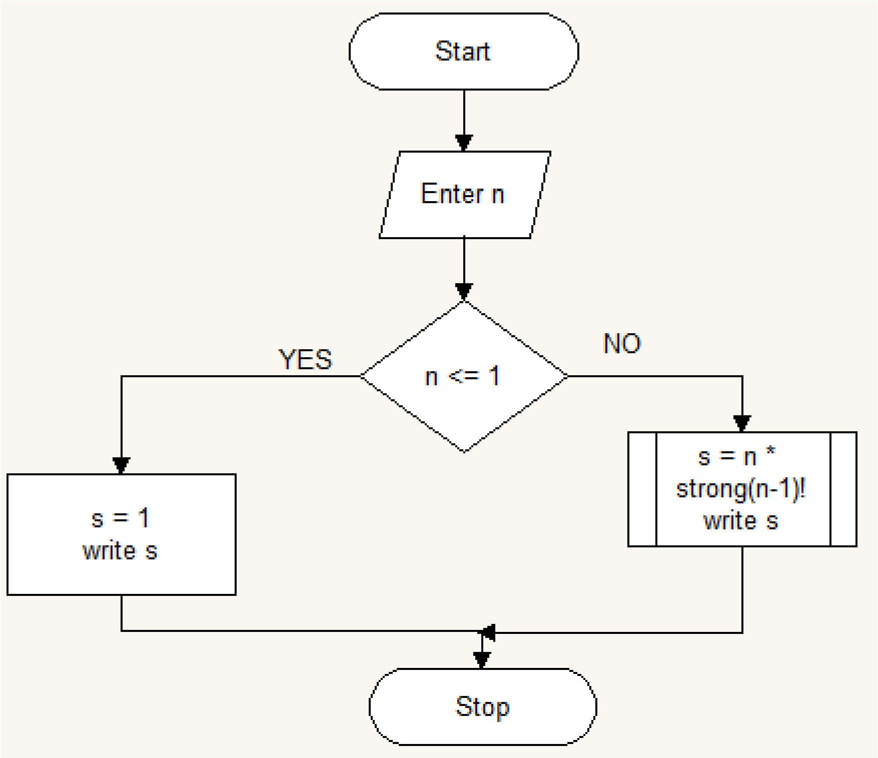
n! = 1 * 2 * 3 * 4 * 5 * 6   6! = 720

There are two ways to compute strong. It can be iterative and recursive. This is the iterative version in the form of a block diagram.



Drawing 9: Interative version [own study]

The recursive version is slightly different.



Drawing 10: Recursive version [own study]

Interface description

The interface is a console pane. The operation of the program is based on user communication. He gives the needed value on the input. The action is short in time. The course of the process and possible operating errors are described in the chapter „Describing of the application's operation".

Source code description

The project was made in the Pascal programming language, in the Dev-Pascal programming environment. All work was done on the Windows 10 operating system. The application's source code is relatively short. It looks like this.

```
Program StrongP;
uses Crt;

var n,i,j:Integer;

begin
    write('Enter n: ');
    readln(n);
    j:=1;
    for i:=1 to n do
```

```
       j:=j*i;
    writeln(#13#10,n,'!=',j);
    ReadKey;;
end.
```

Listing 1: Source code [own study]

The variables i, j, n of the Integer type have been declared. In the main part of the code, a message is displayed to specify the value of n. Then the value is loaded. Then j, which is the result of the program, gets the value 1. Finally, the result is calculated in the for loop. Variable i is a help. Finally, a message is displayed with the result, and "#10#13" gives you a free line for a nice console look. The application waits for any key to be pressed to terminate its work. In Pascal, whatever is between begin and end is always done. Variables are declared earlier.

List of drawings

List of listings