

# C Sharp - DelLast

Program służy do usuwania pierwszych katalogów wg listy alfabetycznej. Czyli mamy katalogi: 1, 2, 3, 4, 5 - i chcemy, aby zostały trzy ostatnie, więc wpisujemy:

```
dellast.exe -n 3 c:\path
```

czyli katalogi 1 i 2 zostaną usunięte.

Użycie:

Usage:

```
dellast.exe -n=n [-test] path
```

Example:

```
dellast.exe -n=25 -test d:\tmp
```

```
dellast.exe -n=15 d:\tmp
```

Opcja -test wylistuje nam katalogi do usunięcia, ale ich nie usunie. Aplikację głównie wykorzystuje w trzymaniu historii archiwów na dysku. Aplikacja jest darmowa i można ją używać za darmo.

Download: źródła binaria – program wymaga bibliotek DotNet 2.0

Kod źródłowy:

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Text;
using System.Text.RegularExpressions;

namespace DelLast
{
    class Program
    {
        static void Main(string[] args)
        {
            //args = new string[2] { "-n=4", "I:\\\" };
            bool test = false;
            int n = 0;
            string folder = "";
            foreach (string arg in args)
            {
                if (arg.Length > 2)
                {
                    if (arg == "-test")
                    {
                        test = true;
                    }
                    else if (arg.Substring(0, 3) == "-n=")
                }
            }
        }
    }
}
```

```

        {
            n = int.Parse(arg.Substring(3, arg.Length - 3)) - 1;
        }
        else
        {
            folder = arg;
        }
    }
}
if (folder == "" || n == 0)
{
    Console.WriteLine("Usage:\n      dellast.exe -n=n [-test]
path\nExample:\n      dellast.exe -n=25 -test d:\\tmp\n      dellast.exe -n=15
d:\\tmp");
    System.Environment.Exit(0);
}
Regex r = new Regex(@"^(([a-zA-Z]
:\:)|(\\))(\{1}|((\{1})[^\{1}][^/*?<>""|]*))+)$");
if (r.IsMatch(folder))
{
    try
    {
        //directoryString = directoryString.Remove (i,
directoryString.Length - i);
        Console.WriteLine("Directories under \"{0}\" are...",
folder);

        string[] directories = Directory.GetDirectories(folder);
        Array.Sort(directories);
        Array.Reverse(directories);
        int i = 0;
        foreach (string dir in directories)
        {
            if (i > n)
            {
                if (test)
                {
                    Console.WriteLine("Subdirectory to del:
 \"{0}\"", dir);
                }
                else
                {
                    Console.WriteLine("Delete dir: \"{0}\"",
dir);
                    Directory.Delete(dir, true);
                }
            }
            i++;
        }
    }
    catch (System.UnauthorizedAccessException)

```

```
        {
            Console.WriteLine("Access denide to folder!");
        }
        catch (System.IO.PathTooLongException)
        {
            Console.WriteLine("To long path!");
        }
        catch (System.IO.DirectoryNotFoundException)
        {
            Console.WriteLine("Path not found!");
        }
        finally
        {
        }
    }
else
{
    Console.WriteLine("Error: " + folder + " - is not folder!");
}
}
}
```

From:

<https://kamil.orchia.pl/> - **kamil.orchia.pl**

Permanent link:

<https://kamil.orchia.pl/doku.php?id=c-sharp-dellast&rev=1364302034>

Last update: **2018/07/16 11:47**

