

Ćwiczenia - wzorki

- Krzysztof Molenda
- wersja 0.1, 2019-03-30

Celem ćwiczenia jest trening/kształcenie w zakresie używania pętli, głównie `for(; ;)`.

Napisz funkcję, która dla zadanego parametru, wypisze na konsoli stosowny wzorek. Funkcja ma mieć własność „skalowalności”, tzn. dla różnych wartości parametrów ma wypisywać różnej wielkości, ale poprawnie wyglądające wzorki. Przy wypisywaniu można użyć jedynie następujących poleceń:

```
Console.Write("*"); Console.WriteLine("*"); Console.Write(" "); Console.WriteLine(" "); Console.WriteLine();
```

Polecenia te, dla wygody, opakowane zostały metodami:

```
Star(), StarLn(), Space(), SpaceLn(), NewLine() :
```

```
const char CHAR = '*';
static void Star() => Console.Write(CHAR);
static void StarLn() => Console.WriteLine(CHAR);
static void Space() => Console.Write(" ");
static void SpaceLn() => Console.WriteLine(" ");
static void NewLine() => Console.WriteLine();
```

❶ „Rysowanie” na konsoli odbywa się w podobny sposób jak wydruk na drukarce igłowej. Rysujemy linię od lewej do prawej, a następnie przechodzimy do nowego wiersza (bez możliwości powrotu do wcześniej narysowanych linii).

Przykład 1

Napisz funkcję, która dla zadanych parametrów `n`, `m` określających rozmiary prostokąta wyrysuje na konsoli jego obraz za pomocą gwiazdek (*) oraz spacji:

Rozwiązanie przykładu 1 (kod)

```
class Program
{
    const char CHAR = '*';
    static void Star() => Console.Write(CHAR);
    static void StarLn() => Console.WriteLine(CHAR);
    static void Space() => Console.Write(" ");
    static void SpaceLn() => Console.WriteLine(" ");
    static void NewLine() => Console.WriteLine();

    public static void Prostokat( int n, int m )
    {
        for( int i = 0; i<n; i++)
        {
            Star();
        }
        NewLine();

        for (int j = 1; j < m-1; j++)
        {
            Star();
            for (int i = 1; i < n - 1; i++)
                Space();

            StarLn();
        }

        for (int i = 0; i < n; i++)
        {
            Star();
        }
        NewLine();
    }
}
```

```

    }

    public static void Main(string[] args)
    {
        Prostokat(5, 7);
    }
}

```

```

*****
* *
* *
* *
* *
* *
*****

```

Przykład 2

Napisz funkcję, która wykreśli na konsoli za pomocą gwiazdek (*) oraz spacji literę x dla zadanego parametru n określającego jej wysokość.

Przykładowy wynik dla $n = 5$:

```

* *
 * *
  *
 * *
  * *

```

Ze względu na kształt litery (linie skośne) zadanie jest trudniejsze od poprzedniego.

1. Dla $n < 3$ rysowanie X nie ma sensu

```
if (n < 3) throw new ArgumentException("zbyt mały rozmiar");
```

2. Dla n parzystych X nie będzie ładnie się prezentowało. Gdy n będzie nieparzyste, linie skośne przetną się w jednym punkcie. Możemy przyjąć, że w przypadku, gdy użytkownik wprowadzi liczbę parzystą, my przekształcimy ją w najbliższą jej liczbę nieparzystą

```
if (n % 2 == 0) n = n + 1;
```

3. Uwaga: $n/2$ jest dzieleniem całkowitym (tzn. $7/2$ daje 3, odcina resztę). Aby wyznaczyć resztę stosujemy operator % (np. $7\%2$ daje 1).

Rozwiązanie przykładu 2 (kod częściowy)

```

const char CHAR = '*';
static void Star() => Console.Write(CHAR);
static void StarLn() => Console.WriteLine(CHAR);
static void Space() => Console.Write(" ");
static void SpaceLn() => Console.WriteLine(" ");
static void NewLine() => Console.WriteLine();

static void LiteraX(int n)
{
    if (n < 3) throw new ArgumentException("zbyt mały rozmiar");
    if (n % 2 == 0) n = n + 1;

    //górna połówka
    for (int i = 0; i < n / 2; i++)
    {
        for (int j = 0; j < i; j++)
            Space();
        Star();
        for (int j = 0; j < n - 2 - 2 * i; j++)
            Space();
        StarLn();
    }
}

```

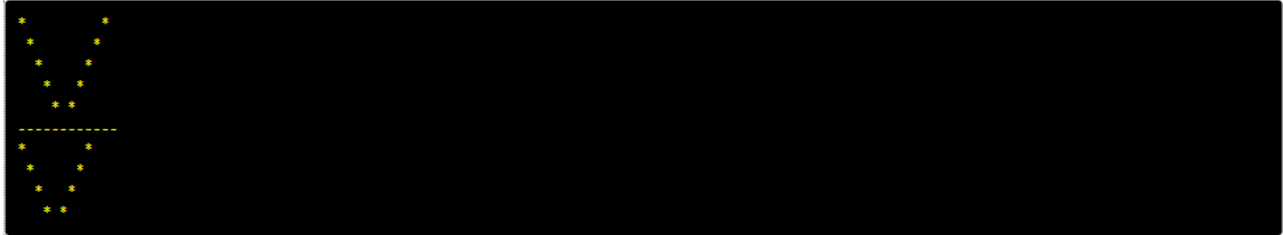
```

        //pojedyncza gwiazdka w środku

        //dolna połówka, symetrycznie do górnej
    }

    static void Main(string[] args)
    {
        LiteraX(11);
        Console.WriteLine("-----");
        LiteraX(8);
    }

```



Zadania do samodzielnego rozwiązania:

⚠ Uwaga: rozwiązywać tak, aby dla różnych wartości parametru n otrzymywać podobny kształt.

Odbicie lustrzane litery **z**

$n = 8$

```

*****
 *
 *
 *
 *
 *
 *
 *
*****

```

Litera **z**

$n = 8$

```

*****
 *
 *
 *
 *
 *
 *
 *
*****

```

Klepsydra

$n = 8$

```

*****
 *  *
 *  *
 **
 **
 *  *
 *  *
*****

```

$n = 7$

```
*****
*  *
*  *
*
*  *
*  *
*****
```

Litera x

n = 8

```
*      *
*      *
*      *
*      *
*      *
*      *
*      *
*      *
```

n = 7

```
*      *
*      *
*      *
*
*      *
*      *
*      *
*      *
```

Trójkąt pusty 1

n = 7

```
*
* *
* *
*****
```

n = 8 (jeśli n parzyste, to tak samo jak dla n-1)

```
*
* *
* *
*****
```

Trójkąt pusty 2

n = 7

```
*****
*  *
*  *
*
```

n = 8 (jeśli n parzyste, to tak samo jak dla n-1)

```
*****
*  *
*  *
*
```

Trójkąt wypełniony

n = 7

```
*****
*****
***
*
```

n = 8 (jeśli n parzyste, to tak samo jak dla n-1)

```
*****
*****
***
*
```

Romb pusty-pełny

n = 7

```
 *
* *
* *
*****
*****
***
*
```

n = 8 (jeśli n parzyste, to tak samo jak dla n-1)

```
 *
* *
* *
*****
*****
***
*
```

Romb pełny-pusty

n = 7

```
 *
****
*****
*****
* *
* *
*
```

n = 8 (jeśli n parzyste, to tak samo jak dla n-1)

```
 *
****
*****
*****
* *
* *
*
```

Romb pełny-pełny

n = 7

```
 *
***
*****
```

```
*****
*****
****
*
```

n = 8 (jeśli n parzyste, to tak samo jak dla n-1)

```
*
***
*****
*****
*****
***
*
```

Litera N

n=8

```
*      *
**     *
* *    *
* * *  *
* * *  *
*  * * 
*   * *
*    **
*     *
```

n=7

```
*      *
**     *
* *    *
* * *  *
*  * * 
*   * *
*    **
*     *
```

Litera N - odbicie lustrzane

n=8

```
*      *
*     **
*   * *
* * * *
* * * *
* *   *
**    *
*     *
```

n=7

```
*      *
*     **
*   * *
* * * *
* *   *
**    *
*     *
```

Szachownica

n=8

```
* * * *
```

```
* * * *
* * * *
* * * *
* * * *
* * * *
* * * *
* * * *
```

Kratownica 1

n=8

```
* * * *
*****
* * * *
*****
* * * *
*****
* * * *
*****
```

Kratownica 2

n=8

```
* * *
*****
* * *
*****
* * *
*****
* * *
*****
```

Litera **κ**

n=8

```
* *
* *
* *
**
**
* *
* *
* *
```

n=7

```
* *
* *
* *
**
* *
* *
* *
```

Litera **v** wąskie

n=8

```
* *
* *
* *
* *
* *
* *
```

```

*
*
```

n=7

```

*  *
*  *
*  *
*  *
*  *
*
*
```

Trójkąt 1

n=8

```

*****
*      *
*      *
*      *
*      *
*      *
*      *
*      *
*      *
```

Trójkąt 2

n=8

```

      *
     **
    ***
   ****
  *****
 *****
*****
```

Litera s

n=8

```

*****
*
*
*****
      *
      *
*****
```

n=7

```

*****
*
*
*****
      *
      *
*****
```

Litera p

n=8

```

*****
```



```
*      *
*      *
*****
*
*
*
```

n=5

```
***
*  *
***
*
*
```

Litera R

n=8

```
*****
*      *
*      *
*****
*  *
*  *
*  *
*      *
```

n=5

```
***
*  *
***
*  *
*  *
```

Litera B

n=8

```
*****
*      *
*      *
*****
*      *
*      *
*****
```

n=5

```
***
*  *
***
*  *
***
```

Cyfra 7

n = 8

```
*****
      *
      *
      *
      *
      *
      *
      *
```

*

Cyfra 3

n = 8

```
*****
      *
      *
*****
      *
      *
*****
```

n=5

```
***
  *
***
  *
***
```

Cyfra 4

n = 8

```
  *
 **
*  *
*  *
****
  *
  *
  *
```

n=7

```
  *
 **
*  *
****
  *
  *
  *
```

Cyfra 5

n=8

```
*****
 *
 *
*****
      *
      *
 *  *
*****
```

Cyfra 6

n=8

```
*****
 *
 *
*****
```

```
*   *  
*   *  
*   *  
*****
```

Cyfra 9

n=8

```
*****  
*       *  
*       *  
*       *  
*****  
      *  
      *  
*       *  
*****
```