

Zadania – Programowanie strukturalne (powtórka z C)

Zadanie 1. Napisz funkcję `Figura(int n)`, która dla podanej przez użytkownika długości boku n ($n \geq 3$) rysuje semigraficznie poniższą figurę. Poniżej przykład dla $n=7$:

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

Do wypisywania linii znaków użyj poleceń:

```
Console.Write("* ");
Console.WriteLine();
Console.WriteLine("Podaj n: ");
Console.Write("Podaj n: ");
```

Do wczytywania znaków użyj poleceń:

```
string element = Console.ReadLine();
string element = Console.Read();
char c = Console.ReadKey();
```

Zadanie 2. Napisz funkcję `Liczby()`, która pobiera od użytkownika n liczb całkowitych (n podaje użytkownik), a następnie oblicza i wypisuje ich średnią oraz wartość maksymalną. Średnią należy wypisać z dwoma miejscami po przecinku (zob. np. <https://docs.microsoft.com/pl-pl/dotnet/standard/base-types/standard-numeric-format-strings>). **Uwaga:** Zadanie zrób bez wykorzystania tablic.

Do parsowania znaków na liczbę całkowitą użyj funkcji : `Int32.TryParse(Console.ReadLine(), out n)`, która zwraca wartość `True` w przypadku podania poprawnej liczby i wartość sparsowaną w zmiennej `n`.

W domu przetestuj obsługę wyjątków i konstrukcję:

```
try
{
    x = Int32.Parse(s);
}
catch (FormatException)
{
    Console.WriteLine("Wymagana jest liczba.");
}
```

(zob. np. https://4programmers.net/C_sharp/Wprowadzenie/Rozdzia%C5%82_8)

Zadanie 3. Napisz funkcję `IleRazy(int [] tab, int a)`, która pobiera tablicę liczb całkowitych oraz dodatkową liczbę całkowitą i zwraca liczbę wystąpień tej liczby w tablicy. Aby przetestować funkcję napisz program `Test()`, który wypełnia n -elementową (n podaje użytkownik) tablicę liczbami całkowitymi z przedziału $[-100, 100]$, pobiera od użytkownika dodatkową liczbę i wywołuje funkcję `IleRazy`.

Do losowania liczb wykorzystaj klasę `Random` w postaci

1. Inicjalizacji generatora : `Random rnd = new Random();`
2. Losowania kolejnej liczby : `tab[i] = rnd.Next(-100, 100);`

(Więcej na stronie: [https://msdn.microsoft.com/pl-pl/library/system.random\(v=vs.110\).aspx](https://msdn.microsoft.com/pl-pl/library/system.random(v=vs.110).aspx)).

Zwróć uwagę na deklarację tablicy postaci : `int[] tab = new int[20];` na długość tablicy `tab.Length`, a także na możliwość wykorzystania dwóch pętli: `for(int i = 0; i < tab.Length; i++)` oraz `foreach (int i in tab)`.

Zadanie 4. Napisz funkcję `Kalkulator()`, która pobiera od użytkownika liczbę, znak działania ('+', '-', '*', '/') oraz drugą liczbę (zakładamy, że poszczególne elementy wyrażenia są oddzielone spacją) i wypisuje wynik działania. Operacje przeprowadzane są na liczbach rzeczywistych, a wynik powinien być wypisany w następującej postaci:

```
10.123
+
3.456
=
13.579
```

Do wyświetlania wykorzystaj standardowe ciągi formatujące dla metody `WriteLine()` lub `Format()` postaci:

```
Console.WriteLine("Wynik wynosi {0,10:F3}", x);
Console.WriteLine("{0,10}", "=");
```

(zob. np. <https://docs.microsoft.com/pl-pl/dotnet/standard/base-types/standard-numeric-format-strings>, lub http://www.java2s.com/Tutorial/CSharp/0040__Data-Type/FormatnumberCD9EF3NXx.htm)

Zadania – Programowanie strukturalne (powtórka z C)

Do wczytania działania wykorzystaj podział na 3 – elementową tablicę rozdzielając ciąg znaków na trzy ciągi względem spacji:

```
string dzialanie = Console.ReadLine();  
string[] el = dzialanie.Split(' ');
```

Zadanie 5. Napisz i przetestuj funkcję `Odwrotnosc(double x)`, która pobiera liczbę rzeczywistą i zwraca jej odwrotność. W przypadku podania liczby 0 wyrzucany jest wyjątek `ArithmeticException`, który trzeba obsłużyć w głównym programie.

Wyjątek należy najpierw zdefiniować:

```
if (x == 0) throw new ArithmeticException("Dzielenie przez 0!!!");
```

Następnie obsłużyć

```
try  
{  
}  
catch(ArithmeticException e)  
{ Console.WriteLine(e.Message); }
```

(zob. np. https://4programmers.net/C_sharp/Wprowadzenie/Rozdzia%C5%82_8)

Zadanie 6. Napisz i przetestuj funkcję `OdwrocZdanie(string zdanie)`, która pobiera zdanie i zwraca wyrazy w kolejności odwróconej. Np. „Umiem programować w języku C#” daje „C# języku w programować Umiem”. **Wskazówka:** Do wydzielenia wyrazów użyj metody `Split()` klasy `String`. (patrz zadanie 4).

Zadania domowe

Zadanie 1

Napisz metodę, która wczytuje liczbę typu `decimal` oraz liczbę rzeczywistą pojedynczej precyzji i wypisuje wynik operacji dodawania, odejmowania, mnożenia oraz dzielenia tych liczb z dokładnością do dwóch miejsc po przecinku.

Zadanie 2

Napisz metodę sprawdzającą, czy podana liczba (dodatnia liczba całkowita) zawiera cyfrę 3.

Zadanie 3

Napisz metodę, która dla liczby dodatniej zwraca jej cyfrowy pierwiastek. Z Wikipedii – pierwiastek **cyfrowy to wartość uzyskana w iteracyjnym procesie sumowania cyfr, przy każdej iteracji do obliczenia sumy cyfr wykorzystywany jest wynik z poprzedniej iteracji. Proces jest kontynuowany aż do osiągnięcia jednocyfrowej liczby.**

Zadanie 4

Mając niepusty tekst, napisz metodę, która zwraca go w formacie skompresowanym. Przykładowe wejście i wyjście:

"kkkkktttttttttt" → "k4t3r10"

"p555ppp7www" → "p153p371w3"

Zadanie 5

Napisz program `Dni()`, który pobiera o użytkownika datę urodzenia, a następnie oblicza i wypisuje ile dni przeżył do dnia dzisiejszego. **Wskazówka:** Do pobrania daty wykorzystaj metodę `Parse()` klasy `DateTime`. Bieżącą datę odczytasz wykorzystując właściwość `Now()` tej samej klasy.

Zadanie 6

Napisz program, który podaje ile dni pozostało do końca roku licząc od dnia dzisiejszego.

Zadanie 7

Napisz program `Wyświetl()`, który wyświetla zawartość dowolnego pliku tekstowego na ekranie. Plik do wyświetlenia podawany jest jako argument funkcji `Main()`. **Wskazówka:** Do odczytu wszystkich linii pliku tekstowego do tablicy `string[]` możesz wykorzystać metodę `System.IO.File.ReadAllLines()`.