

Programowanie pod Windows

Zestaw 1

Język C# - rozgrzewka

2023-02-28

Liczba punktów do zdobycia: **6/6**

Zestaw ważny do: 2023-03-14

1. (**1p**) Napisać program, który wyznacza zbiór wszystkich liczb naturalnych 1 a 100000, które są podzielne zarówno przez każdą ze swoich cyfr z osobna jak i przez sumę swoich cyfr.
2. (**1p**) Przygotować rozwiązanie (Solution) które składa się co najmniej z czterech projektów (Project): dwu aplikacji konsolowych i dwu bibliotek.

W każdej z bibliotek umieścić po jednej klasie z jedną metodą. Dodać referencje do bibliotek z każdej aplikacji konsolowej. Nie dodawać referencji pomiędzy aplikacjami konsolowymi.

W każdej z aplikacji konsolowych napisać fragment kodu, który wywołuje kod z obu bibliotek.

Pokazać jak z poziomu Visual Studio uruchomić jedną z aplikacji konsolowych, potem drugą (Set as startup project...) a potem obie naraz.

Pokazać jak w każdym z tych sposobów uruchomienia kodu można umieszczać pułapki w kodzie i debugować kod.

3. (**1p**) Zdokumentować (przez umieszczenie odpowiednich komentarzy w kodzie) jeden dowolny program z bieżącej sekcji.

Wygenerować dokumentację w postaci pliku XML podczas kompilacji. Użyć narzędzia SandCastle Help File Builder (<https://github.com/EWSoftware/SHFB>) do zbudowania pomocy w obsługiwanych przez SandCastle stylach (np. Website).

4. (**1p**) Napisać w C# dowolny program demonstrujący użycie klas (metod, pól, właściwości, indeksów, delegacji i zdarzeń) oraz podstawowych konstrukcji składniowych (pętle, instrukcje warunkowe, `switch`) i zdekompilować go za pomocą narzędzia **ILSpy** (<http://ilspy.net/>).

Otrzymany kod skompilować (ilasm), aby otrzymać plik wynikowy. Plik ten następnie zdekompilować na powrót do języka C#.

Porównać otrzymane w ten sposób pliki z kodem źródłowym. Jak objawiają się i z czego wynikają różnice?

5. (**2p**) Zaimplementować klasę siatki dwuwymiarowej, `Grid`, z dwoma indeksami:

- jednowymiarowym, zwracającym listę elementów zadanego wiersza tablicy, tak aby klient klasy mógł napisać:

```
...
Grid grid = new Grid( 4, 4 );
int[] rowdata = grid[1]; // akcesor "get"
```

- dwuwymiarowym, zwracającym określony element tablicy, tak aby klient klasy mógł napisać:

```
...
Grid grid = new Grid( 4, 4 );

elem[2, 2] = 5;          // akcesor "set"
int elem = grid[1, 4]; // akcesor "get"
```

Oba indeksy powinny przyjmować jako parametry liczby całkowite. Konstruktor klasy powinien przyjmować jako parametry liczbę wierszy i liczbę kolumn siatki.

Wiktor Zychla