

# P3Z - lab05

## grupa A

Zapisać liczbę  $n$  w systemie pozycyjnym o podstawie  $b$  nazywamy ciąg

$$c_k c_{k-1} c_{k-2} \dots c_2 c_1 c_0$$

gdzie  $n = c_0 b^0 + c_1 b^1 + c_2 b^2 + \dots + c_k b^k$  i  $\forall_{i \in \{1, \dots, k\}} c_i \in \{0, 1, \dots, b-1\}$

Dla  $b > 10$  do zapisu liczb używamy kolejnych liter alfabetu angielskiego: ABCDEFGHIJKLMNOPQRSTU-VWXYZ

### Część 1 (1 pkt.)

Stworzyć klasę *NumberInBase* reprezentującą zapis liczby w wybranym systemie liczbowym. Klasa posiada dwa pola:

- $b$  typu `int`, oznaczające podstawę, w której zapisana jest liczba,
- $number$  typu `string`, oznaczające zapis liczby w systemie o podstawie  $b$ .

Napisać konstruktory:

- *NumberInBase()* - konstruktor bezparametrowy, tworzący liczbę 0 w systemie dziesiętnym
- *NumberInBase(int n)* - tworzący liczbę  $n$  w systemie dziesiętnym
- *NumberInBase(string s, int b)* - tworzący liczbę o zapisie  $s$  w systemie o podstawie  $b$

Dodać metodę *Print()* zwracającą `string` z zapisem liczby w formacie (number)\_b, np. dla 100101 w systemie binarnym zwrócić (100101)\_2, dla 129 w systemie dziesiętnym zwrócić (129)\_10 itp.

### Część 2 (1 pkt.)

Stworzyć klasę *NumbersConverter* do konwersji liczb między systemami liczbowymi. Klasa posiada jedno pole  $bases$  typu `int[][]`.

Napisać konstruktor bezparametrowy, w którym wypełniane jest pole  $bases$  potęgami kolejnych liczb, tak aby w  $bases[k][i]$  znajdowała się wartość  $i$ -ta potęga liczby  $k$ , czyli  $k^i$ . Wypełniamy dla potęg od 2 do 30 aż wartość dla każdej z nich nie przekroczy 1000000 (wskazówka: można użyć `Math.Log(a,b)`, która zwraca logarytm z  $a$  przy podstawie  $b$ ).

### Część 3 (1.5 pkt.)

Dodać metodę *ConvertToDecimal* w klasie *NumbersConverter* przyjmującą parametr typu *NumberInBase*. Metoda powinna zwracać liczbę typu `int` - przekonwertowaną na system dziesiętny liczbę podaną w parametrze.

### Część 4 (1.5 pkt.)

Dodać metodę *ConvertFromDecimal* w klasie *NumbersConverter* przyjmującą dwa parametry typu `int`:  $n$  i  $b$ . Metoda powinna zwracać obiekt typu *NumberInBase* z zapisem liczby  $n$  o podstawie  $b$ .

Punktacja części 3 i 4: tylko system binarny - 0.5pkt., konwersja systemów o podstawach 2-10 - 1pkt., konwersja systemów o podstawach 2-30 - maksymalna liczba punktów (1.5pkt.).