

Programowanie Java
Anna Gogolińska
Zestaw 1

Zad 1. Napisz pierwszy program „Witaj świecie”. Uruchom go z linii poleceń.

Zad 2. Napisz program, który wypisze na ekran ilość argumentów z którymi został uruchomiony oraz każdy z tych argumentów – w osobnej linii. Uruchom go z linii poleceń i w IntelliJ.

Zad 3. Zaimplementuj klasę *TabliczkaMnozenia*, która podczas uruchomienia pobierze liczbę n podaną jako jedyny argument (z linii poleceń), a następnie wypisze na ekran tabliczkę mnożenia liczb od 1 do n . Zmodyfikuj klasę, aby n było pobierane ze standardowego wejścia.

Zadania Samodzielne

Zad 4. Napisać klasę *Calkowanie*, która będzie posiadała metodę *obliczKwadratowe* wyliczającą wartość całki oznaczonej dla równania kwadratowego. Program podczas uruchomienia powinien pobrać wartości 5 argumentów: a , b , c , $x1$ i $x2$. Pierwsze trzy liczby oznaczają współczynniki równania kwadratowego $ax^2 + bx + c$, natomiast dwie pozostałe granice całkowania (od $x1$ do $x2$). Wynikiem powinna być liczba będąca przybliżoną wartością całki oznaczonej z podanego równania kwadratowego.

Zad 5

Napisz program, który wczyta jeden wiersz tekstu i wydrukuje go w wersji zaszyfrowanej. Szyfrowanie pojedynczej litery polega na odpowiedniej zmianie kolejności czterech najmniej znaczących bitów. Miejscami powinny zostać zamienione trzeci bit z czwartym oraz pierwszy z drugim. Program ma wczytywać ze strumienia wejściowego znaki tak długo aż napotka znak nowej linii. Wówczas ma wypisać zakodowany ciąg znaków.

Przykład:

Wejście:
x&15

Wyjście:
t)2:

Zad 6

Napisz program, który czyta ze standardowego wejścia najpierw dwie liczby zmiennoprzecinkowe, a następnie dwa słowa. Program ma następnie na ekran wypisać tekst (dwie linie):

Liczba pierwsza: *tutaj_pierwsza_liczba*, liczba druga: *tutaj_druga_liczba*.

Pierwszy napis: *tutaj_pierwsze_slowo*, drugi napis: *tutaj_drugie_slowo*.

Liczby mają być wypisane z dokładnością do 3 miejsc po przecinku na 10 pozycjach, wyrównane do lewej. Należy użyć klasy *Formatter* oraz ewentualnie *StringBuilder*.