

## Równanie kwadratowe

---

### Cel ćwiczenia

Celem ćwiczenia jest narysowanie wykresu funkcji kwadratowej  $y = a * x^2 + b * x + c$  w zakresie od -10 do +10 , za pomocą krótkich odcinków wyświetlanych w okienku graficznym wygenerowanym z pomocą biblioteki *StdDraw* . Program ma samodzielnie ustalić skalę na osi OY tak, aby na rysunku został wyświetlony wykres w całym zakresie zmiennej x .

### Wykonaj po kolei poniższe polecenia

1. Stwórz nowy projekt. Możesz go nazwać *RownanieKwadratowe*. Dodaj pakiet i klasę o tej samej nazwie.
2. Do projektu dodaj bibliotekę *StdDraw* z pliku *StdDraw.java* który stworzyłeś wcześniej lub pobierz ją z Sieci, zapisz w pliku i zaimportuj. Popraw nazwę pakietu zawartą w pliku źródłowym biblioteki na własną.

<https://introcs.cs.princeton.edu/java/stdlib/StdDraw.java>

3. Zadeklaruj zmienne a, b i c typu *double* i pobierz ich wartość z konsoli.
4. Zadeklaruj zmienną dx typu *double* i ustal jej wartość na 1. Do deklaracji zmiennej przed deklaracją typu dodaj słowo *final* w celu zablokowania możliwości zmiany wartości tej zmiennej.

```
final double dx = 1;
```

5. W zakresie zmiennej x od -10 do +10 jest  $\frac{-10 - (-10)}{1} + 1 = 21$  wartości. Oblicz te wartości i zapisz je w 21-elementowej tablicy do przechowywania wartości typu *double*.
6. Oblicz wartość największego elementu tej tablicy i zapisz ją w zmiennej *max* typu *double*.
7. Oblicz wartość najmniejszego elementu tej tablicy i zapisz ją w zmiennej *min* typu *double*.
8. Ustaw wielkość okna graficznego na 800 x 600 pikseli. Użyj do tego celu statycznej metody *setCanvasSize(int width, int height)* klasy *StdDraw* .

9. Ustaw zakres osi X na od -12 do +12 za pomocą metody `setXscale(double min, double max)`.

10. Ustawienie zakresu osi OY jest trochę trudniejsze do wytłumaczenia :) .

A więc jeżeli wartość bezwzględna max jest większa niż wartość bezwzględna min to należy ustawić zakres osi OY na od -max do + max, w przeciwnym wypadku należy ustawić zakres osi OY od +min do -min .

Cała ta komplikacja bierze się stąd, że wartości min i max mogą być ujemne. Oczywiście do ustalenia skali użyj statycznej metody `setYscale(double min, double max)` .

BTW: Aby zostawić margines na wykresie do zakresu osi Y dobrze do dolnego i górnego końca skali dodać jeszcze np. 10% wartości.

11. Ustaw kolor pióra na szary, a wielkość na 0,004 .

12. Narysuj oś OX rysując linię od punktu (-11, 0) do punktu (+11,0) korzystając z metody `line(double x1, double y1, double x2, double y2)`.

13. Narysuj skalę na osi OX od -9 do +9 rysując 19 krótkich odcinków od punktu (-0,1 x) do (+0,1 x), gdzie x zmienia się od -9 do 9 co jeden.

14. Narysuj oś OY rysując linię od punktu (0, -z) do punktu (0,+z) korzystając z metody `line(double x1, double y1, double x2, double y2)`, gdzie z jest wartością ustaloną podczas obliczania wartości minimalnej i maksymalnej elementów tablicy y.

15. Sprawdź wygląd obrazka.

16. Zmień kolor pióra na jakiś kolorowy :) za pomocą metody `setPenColor(int red, int green, int blue)` , gdzie red, green i blue określa intensywność każdej ze składowych koloru: czerwonej, zielonej i niebieskiej w skali od 0 do 255.

17. Tym razem wykres narysujemy za pomocą odcinków między kolejnymi punktami o współrzędnych (x, y[i]) i (x+1, y[i+1]). Takich odcinków jest  $\frac{-10 - (-10)}{1} = 20$  i do ich narysowania należy użyć pętli powtarzanej 20 razy oraz statycznej metody `line(double x1, double y1, double x2, double y2)` .

18. Sprawdź wygląd wykresu.

19. Udoskonal wykres zmieniając wartość dx na 0,1 . Odpowiednio zmień także wielkość tablicy współrzędnych y oraz ilość powtórzeń w pętlach wyszukujących wartość maksimum i minimum, rysującej wykres oraz krok podczas zmiany wartości x.

20. Enjoy.