

## Parabola

---

### Cel ćwiczenia

Celem ćwiczenia jest narysowanie wykresu funkcji kwadratowej  $y = x^2$  w zakresie od -10 do +10, mającego postać zbioru pojedynczych punktów wyświetlanych w okienku graficznym wygenerowanym z pomocą biblioteki *StdDraw*.

### Lista poleceń

1. Stwórz nowy projekt. Możesz go nazwać *ParabolaP*. 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. Ustaw wielkość okna graficznego na 800 x 600 pikseli. Użyj do tego celu statycznej metody *setCanvasSize(int width, int height)* klasy *StdDraw*.
4. Ustaw zakres osi X na od -12 do +12 za pomocą metody *setXscale(double min, double max)* oraz osi Y od -20 do +120 za pomocą metody *setYscale(double min, double max)*.
5. Narysuj osi układu współrzędnych. Korzystając z metody *line(double x1, double y1, double x2, double y2)* narysuj oś OX za pomocą linii od punktu (-11, 0) do punktu (+11,0) oraz oś OY za pomocą linii od punktu (0,-10) do punktu (0, +110).
6. Zmień wielkość pióra na 0,02 za pomocą metody *setPenSize(double radius)*.
7. Wykres zostanie narysowany za pomocą 21 punktów, dla kolejnych wartości zmiennej x od -10 co jednostkę, aż do +10 oraz dla wartości zmiennej y obliczonych z równania. W tym celu zadeklaruj pętlę powtarzającą obliczenia 21 razy za pomocą polecenia *for*.
8. W pętli umieść polecenie rysujące pojedynczy punkt o współrzędnych obliczonych w każdym kroku pętli.
9. Sprawdź wygląd wykresu.
10. Zmień kolor osi na własny (inny niż czarny) za pomocą metody *setPenColor(int red, int green, int blue)*, gdzie indywidualne składowe metody oznaczają intensywność

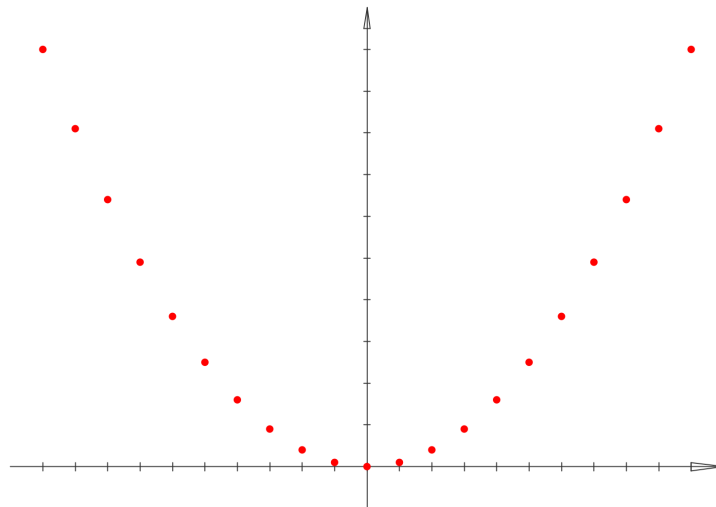
koloru składowego w skali od 0 do 255 (włącznie).

11. Zmień kolor punktów np. na własny (inny niż czarny) czerwony za pomocą tej samej metody.
12. Dodaj do osi groty strzałek. W tym celu na zakończeniu osi OX narysuj trzy linie między punktami: (10; 1), (10; -1) oraz (11;0) — dla osi OX oraz między punktami: (-0,1; 105), (0,1; 105) oraz (1; 110).
13. Do wykresu dodaj skalę osi OX oraz OY.

Aby narysować skalę na osi OX zadeklaruj pętlę wykonującą 21 powtórzeń. Zmieniaj zmienną sterującą  $i$  w zakresie od -10 do +10 (włącznie) . Podczas każdego obrotu pętli rysuj odcinek między punktami o współrzędnych  $(i; -1)$  i  $(i; 1)$ .

Aby narysować skalę na osi OY W tym celu zadeklaruj pętlę wykonującą 11 powtórzeń dla osi OY. Zmienną sterującą  $i$  zmieniaj w zakresie od -1 do 10 (włącznie). Podczas każdego obrotu pętli rysuj odcinek między punktami o współrzędnych  $(-0,1; 10*i)$  i  $(0,1; 10*i)$  .

14. Ciesz się swoim wykresem albo zabierz się za jego udoskonalanie.



Możesz się też nim pobawić np. rysując każdy punkt w innym, losowym kolorze. W tym celu możesz generować trójki liczb z zakresu od 0 do 255 lub element tablicy zawierający kilka predefiniowanych kolorów.

Możesz także dodać buforowanie rysunku w pamięci, dzięki czemu będzie się szybko rysował. W tym celu na początku programu użyj metody `enableDoubleBuffering()` , a na końcu metody `show()` .