



## WYDZIAŁ ELEKTRONIKI I INFORMATYKI

Rok akademicki <b>2021/2022</b>	<b>TEMAT:</b> Program rysujący krzywą balistyczną		
Kierunek studiów:	<b>Informatyka</b>	Wykonawcy: <b>Patryk Banaś</b>	
Semestr:	<b>V</b>		
Grupa:	<b>TliM</b>		

### 1. Cel ćwiczenia

Celem ćwiczenia jest stworzenie komponentu w technologii JavaBeans oraz aplikacji testującej, który będzie rysować krzywą balistyczną na wykresie, dla podanych danych wejściowych.

### 2. Zakres ćwiczenia

- Rozpoczęcie projektu.
- Wykonanie komponentu [BallisticChart].
- Napisanie programu [TestEnv].
- Dodanie komponentu do programu.
- Przetestowanie komponentu.

### 3. Opis ćwiczenia

Komponent ma za zadanie rysować krzywą balistyczną na osi X-Y, dla podanych wartości:

- Początkowe współrzędne  $[x]$ ,  $[y]$
- Kąt  $[\theta]$ , grawitacja  $[g]$  i prędkość  $[v]$
- Wysokości i szerokości okna

Wartości  $y$  jest wyliczana za pomocą wzoru:

$$y = x \tan(\theta) - \frac{x^2 g}{2v^2 \cos^2(\theta)}$$

Wykres automatycznie się dostosowuje do podanej wysokości i szerokości okna, wraz z wartościami na osiach wykresu.

Program został napisany w języku JAVA i zostało wykorzystane środowisko IDE Eclipse.

### 4. Cechy komponentu

- Łatwy w obsłudze.

- Prosty w zaimplementowaniu.
- Dostosowuje zakres osi X-Y do krzywej
- Możliwość modyfikacji layout'u

## 5. Opis funkcjonalny

Nr	Nazwa	Opis	Trudność implementacji* (1-5)	Priorytet* (1-5)
1	Rysowanie wykresu	Inicjalizowanie i rysowanie wykresu w oknie JFrame.	5	5
2	Ustawienie koloru	Ustawienie dowolnego koloru w formacie HSB, dla punktów na wykresie.	1	1
3	Ustawienie zakresu osi XY	Ilość wartości na zakresie X i Y	4	4
4	Przypisywanie etykiet osi XY	Przypisanie etykiet dla osi X i Y	2	3
5	Ustawienie marginesu wykresu	Margines od krawędzi w oknie JFrame.	3	1
6	Ustawienie wielkości punktów	Ustala wielkość każdego punktu na wykresie.	1	2

\*5 oznacza największy priorytet, 1 – najmniejszy

## 6. API

Nr	API	Funkcja
1	<code>public BallisticChart (int x, int y, double gravity, double velocity, double angle, int w, int h)</code>	Rysowanie wykresu. Inicjalizacja wykresu.  <b>Domyślne wartości:</b> x: 0, y: 0, gravity: 0, velocity: 0, angle: 0, w: 400, h: 400
2	<code>public void setFrameOffset (int offset)</code>	Ustawienie marginesu wykresu  <b>Domyślne wartości:</b> offset: 50
3	<code>public void setXYZZoom (int zoom)</code>	Ustawienie zakresu osi XY  <b>Domyślne wartości:</b> zoom: 12
4	<code>public void setPointSize (int size)</code>	Ustawienie wielkości punktów  <b>Domyślne wartości:</b> size: 4
5	<code>public void setXTag (String xtag)</code>	Przypisywanie etykiet osi XY  <b>Domyślne wartości:</b> xtag: „x”

6	<code>public void setYTag (String ytag)</code>	Przypisywanie etykiet osi XY  <b>Domyślne wartości:</b> ytag: „y”
7	<code>public void setColor (int h, int s, int b)</code>	Ustawienie koloru  <b>Domyślne wartości:</b> h: 50, s: 100, b: 100

## 6. Aplikacja testująca

<b>Numer testu</b>	<b>Test 1</b>
<b>Autor</b>	Patryk Banaś
<b>Data</b>	1-12-2021
<b>Nazwa testu</b>	Testowanie komponentu – zmiana koloru punktów
<b>Nazwa testowanej funkcjonalności</b>	Funkcja zmiany koloru punktów
<b>Warunek wstępny</b>	Uruchomienie aplikacji „TestEnv” w OS Windows 10, z komponentem „BallisticCurve”
<b>Wartości wejściowe</b>	Wartości wejściowe: kolor w postaci liczbowej (wartości od 1 do 255)
<b>Kroki</b>	Wprowadzenie wartości koloru. Kliknięcie przycisku „Rysuj”.
<b>Oczekiwany wynik</b>	Narysowanie wykresu z funkcja w kolorze
<b>Warunek końcowy</b>	Narysowanie wykresu. Punktu wykresu we wybranym kolorze.
<b>Kryterium zaliczenia</b>	Wykres funkcji wyświetlony w czasie <5s, punkty o podanym kolorze

<b>Numer testu</b>	<b>Test 2</b>
<b>Autor</b>	Patryk Banaś
<b>Data</b>	1-12-2021
<b>Nazwa testu</b>	Testowanie komponentu – rysowanie wykresu
<b>Nazwa testowanej funkcjonalności</b>	Moduł do rysowania wykresu

<b>Warunek wstępny</b>	Uruchomienie aplikacji „TestEnv” w OS Windows 10, z komponentem „BallisticCurve”
<b>Wartości wejściowe</b>	Wartości wejściowe: <ul style="list-style-type: none"> <li>· Początkowe współrzędne <math>[x]</math>, <math>[y]</math></li> <li>· Kąt <math>[\theta]</math>, grawitacja <math>[g]</math> i prędkość <math>[v]</math></li> <li>· Wysokości i szerokości okna</li> </ul>
<b>Kroki</b>	Podanie random-owych wartości na polach: $[x]$ , $[y]$ , $[\theta]$ , $[g]$ , $[v]$ , $[\text{height}]$ , $[\text{width}]$ . Kliknięcie przycisku „Rysuj”.
<b>Oczekiwany wynik</b>	Otwarcie nowego okna, wyświetlenie paraboli-stycznego wykresu. O kształcie $y=\sin(x)$ dla $x < 0;\pi$
<b>Warunek końcowy</b>	Wyświetlone nowe okno z wykresem o kształcie paraboli
<b>Kryterium zaliczenia</b>	Wykres funkcji wyświetlony w czasie $< 5s$ , kształt funkcji zbliżony do $y=\sin(x)$ dla $x < 0;\pi$ o dowolnej wysokości i szerokości

Wykres powinien zawsze przyjmować ten sam kształt.

<b>Numer testu</b>	<b>Test 3</b>
<b>Autor</b>	Patryk Banaś
<b>Data</b>	1-12-2021
<b>Nazwa testu</b>	Testowanie komponentu – zmiana precyzji wykresu
<b>Nazwa testowanej funkcjonalności</b>	Funkcja zmiany precyzji wykresu
<b>Warunek wstępny</b>	Uruchomienie aplikacji „TestEnv” w OS Windows 10, z komponentem „BallisticCurve”
<b>Wartości wejściowe</b>	Wartości wejściowe: wartość precyzji (lub zoom wykresu), w minimalna wartość 2, maksymalna 50
<b>Kroki</b>	Podanie wartości na polu: precyzja. Kliknięcie przycisku „Rysuj”.
<b>Oczekiwany wynik</b>	Poprawnie wyświetlony zakres na wykresie.
<b>Warunek końcowy</b>	Wyświetlenie zakresu dla podanej wartości.
<b>Kryterium zaliczenia</b>	Wykres funkcji wyświetlony w czasie $< 5s$ , ilość liczb na podanych zakresach XY równa wartości podanej w polu precyzji.

## 7. Protokół założycielski

1. Dnia 6 grudnia 2021 roku utworzony został zespół jednoosobowy którego celem jest utworzenie oprogramowania komponentowego na przedmiot Programowanie komponentów, oraz Przeprowadzenie inżynierii oprogramowania do projektu dla przedmiotu Inżynieria oprogramowania

2. Przewidywana data zakończenia oprogramowania jest 6 grudnia 2021?

3. W skład grupy wchodzi

- Patryk Banaś

4. Tytuł projektu: *Ballistic Chart*

5. Kierownik Projektu

- Patryk Banaś

6. Opis projektu

Projekt zakłada stworzenie komponentu rysującego krzywą balistyczną. Całe założenia powinny zostać zanotowane jak by to był projekt komercyjny. Program trzeba wykonać w Java.

7. Cel projektu:

- Napisanie komponentu rysującego krzywą balistyczną w Java
- Udokumentowanie pracy

## 8. Podsumowanie

Celem wykonania zadania było stworzenie programu, który spełnia założenia komponentu. Zrobiono zadanie zgodnie z założeniami.