

SPRAWOZDANIE

Zajęcia: Grafika komputerowa

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium: 1

Data: 28.02.2023

Temat: "Przekształcenia 2D w bibliotece Java 2D"

Wariant Ośmiokąt, figura nr.4

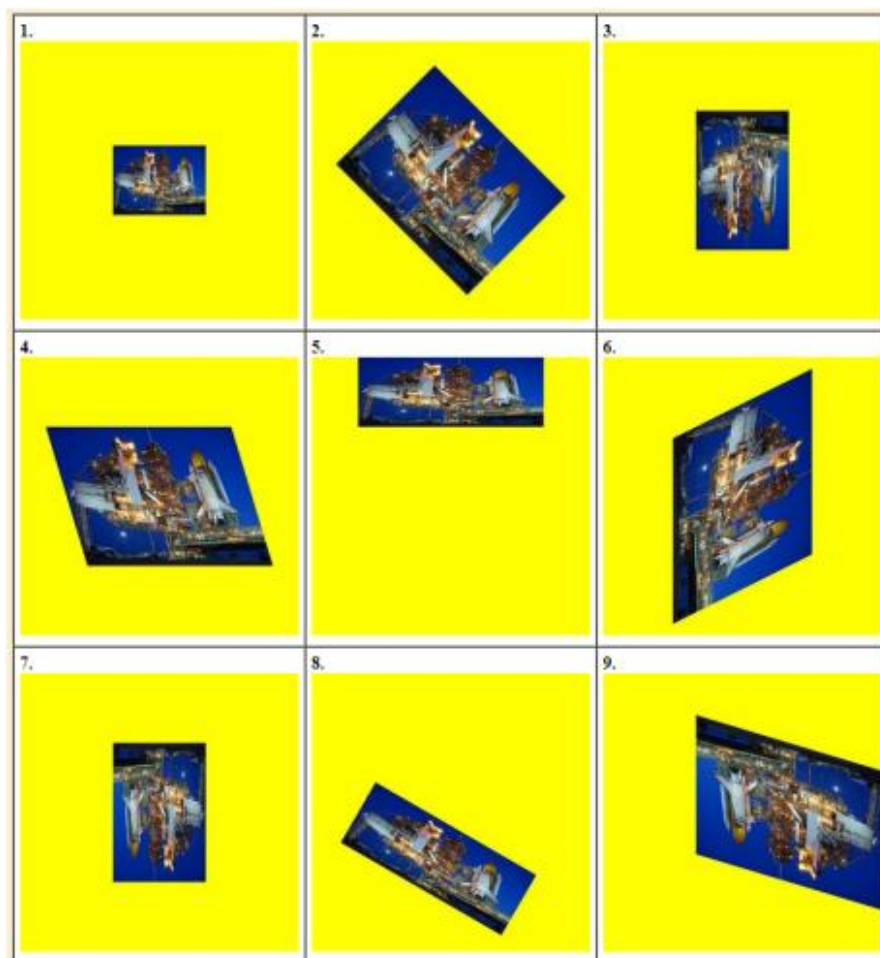
Igor Gawłowicz,
Informatyka I stopień,
stacjonarne,
4 semestr,
Gr.2b

1. Polecenie nr. 1:

Program Transform2D.java rysuje obraz shuttle.jpg w panelu. Narysować zamiast obrazu wielokąt według wariantu (liczba n) w panelu wyświetlania. Panel ma wymiary 600 na 600 pikseli, a wielokąt ma promień 150 pikseli. Okno zawiera również wyskakujące menu z etykieta "Transform:". Opcje w menu to "None" i cyfry od 1 do 9. W tym programie menu wyskakujące nie działa. Zadanie polega na dodaniu kodu do metody paintComponent (). (Miejsce jest oznaczone TODO.) Kiedy wybór ma wartość 0, strona powinna wyświetlać obraz nietransformowany. W przypadku innych możliwych wartości musisz zastosować przekształcenie (lub będziesz potrzebować I kombinacji przekształceń) dla każdej z wartości od 1 do 9 (patrz Fig. 1).

2. Sposób rozwiązania zadania z polecenia nr. 1:

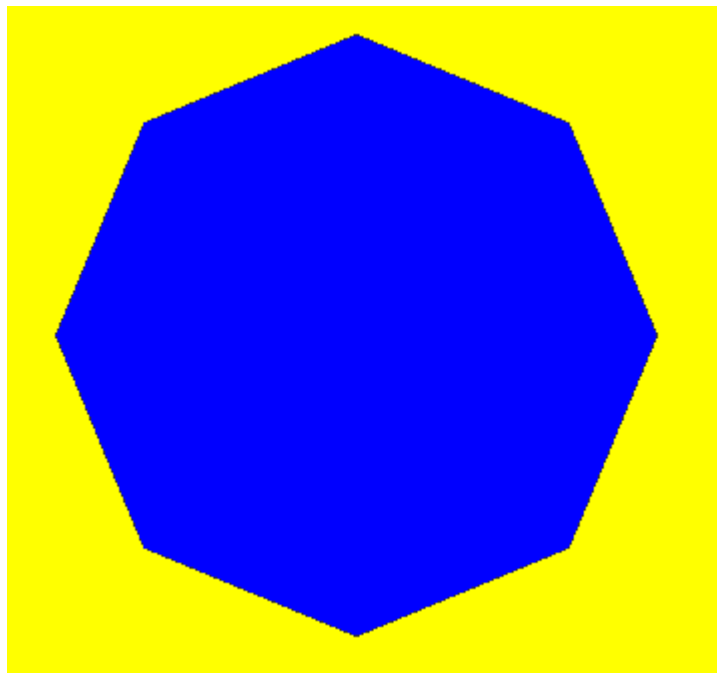
Pierwszym krokiem było przeanalizowanie podanej z zadaniem figury.



Następnie przy użyciu instrukcji **switch**, która za argument przyjęła zmienną **whichTransform** napisałem instrukcje potrzebne do otrzymania oczekiwanych przekształceń takie jak:

- **scale** - skalujący obraz/figurę **x** razy,
- **rotate** – obraca obraz/figurę o daną wartość (przy użyciu funkcji matematycznej **toRadians** byłem w stanie sprecyzować dokładny kąt o jaki chcę obrócić obiekt),
- **shear** – deformuje obraz/figurę względem osi x lub y,
- **translate** – przesuwa obraz/figurę o x i y jednostek,

Kolejnym krokiem było utworzenie oczekiwanej figury, w moim przypadku był to ośmiokąt który stworzyłem dzięki tablicy współrzędnych 8 następnych punktów na kole, aby otrzymać figurę foremną o ośmiu kątach poprzez utworzenie obiektu klasy **Poligon**, który przyjmuję za argumenty w konstruktorze tablicę punktów x i y oraz ilość wierzchołków.



Otrzymana figura w oryginalnym rozmiarze

<Kod źródłowy: [Kod Github zadanie 1](#) >

3. Polecenie nr.2:

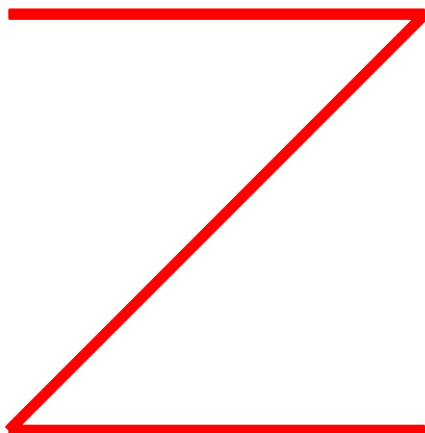
Narysować figurę określoną wariantem (patrz Fig. 2). Dostępne są trzy podstawowe kształty: `circle ()`, `square ()` i `triangle ()`. Zacznij od programu `TransformedShapes.java`. TODO. Możesz użyć poleceń do rysowania, takich jak `g.fillRect ()` itp.

4. Sposób rozwiązania zadania z polecenia nr.2:

Zadanie zacząłem od ustawienia koloru za pomocą instrukcji **g2.setColor(Color.red)** oraz grubości linii za pomocą **g2.setStroke(new BasicStroke(10))** w moim przypadku wylosowałem figurę jak na obrazku poniżej.



Udało mi się wykonać za pomocą 3 osobnych obiektów typu **Polygon**, które przyjęły kształt 3 odcinków łączących punkty w odpowiednich współrzędnych, dając nam figurę w kształcie litery **Z**.



Moja figura dla porównania
<Kod źródłowy: [Kod Github zadanie 2](#) >

5. Wnioski:

Powyższe zadania pomogły mi zrozumieć podstawy budowy klas w języku Java oraz zastosowanie bibliotek **2D Graphics** wraz z jej licznymi funkcjami budowy przeróżnych kształtów.