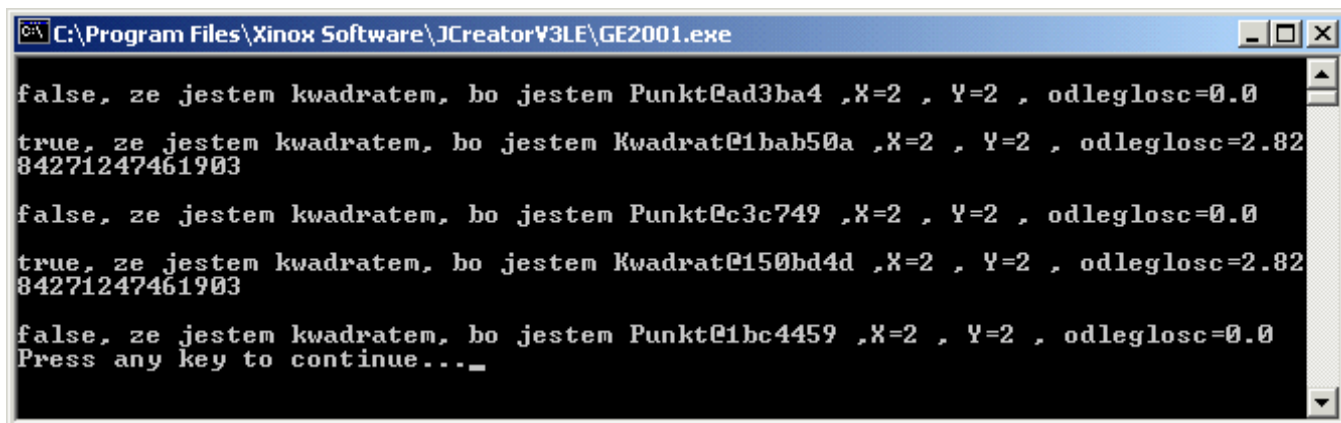


## Laboratorium 6

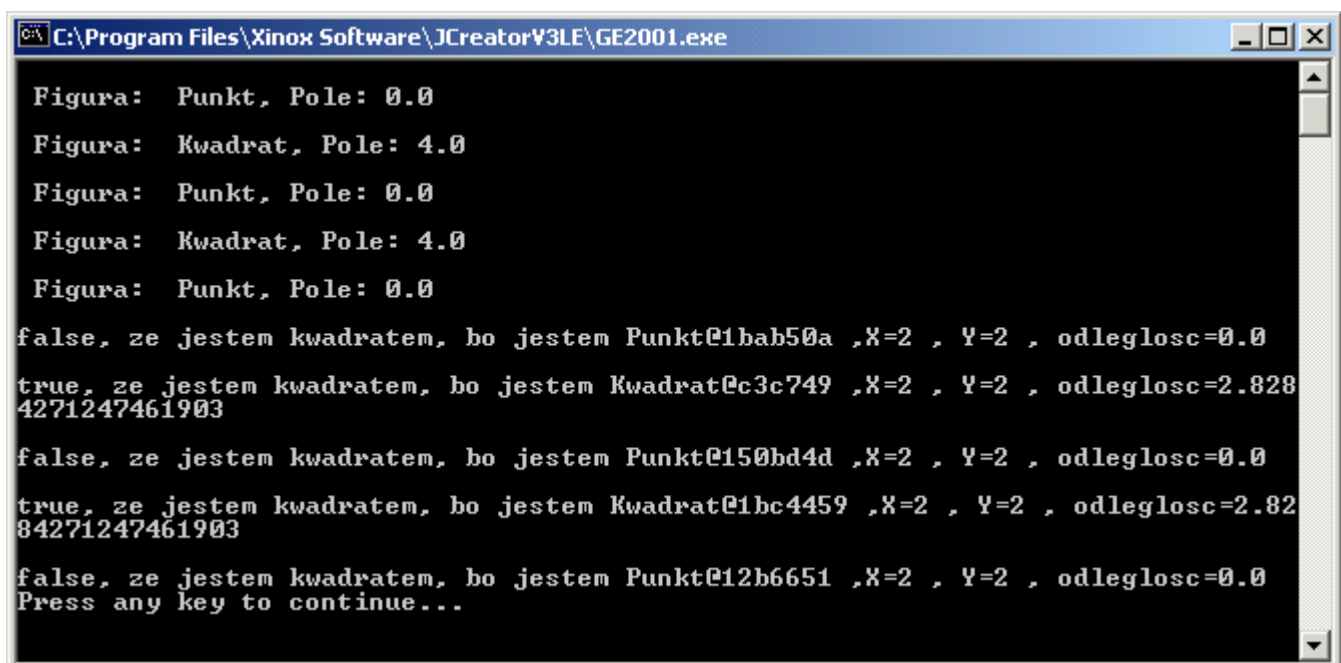
Cel: Budowa klas z wykorzystaniem dziedziczenia oraz implementowania interfejsów. W prowadzenie do trybu graficznego. Animacje obiektów w trybie graficznym.

1. Wykonaj program (klasa *Obraz1\_1*), który wyświetla na ekranie dane figur typu *Punkt*, *Kwadrat* tak, jak program *Obraz1* (projekt *Rys1*) oraz dodatkowo dane prostokąta. Klasa *Prostokat* powinna dziedziczyć od klasy *Kwadrat* (definicja konstruktora w klasie *Prostokat*) i dodatkowo powinna posiadać metody obsługujące drugi bok prostokąta (*szerokosc*). W programie *Obraz1\_1*, który ma teraz dodatkowo definicję klasy *Prostokat* w porównaniu z programem *Obraz1*, należy w klasie *Figury* w metodzie *wypelnij()* dodać wstawianie obiektów typu *Prostokat* do tablicy *figury*. Należy wyjaśnić pochodzenie metody *toString* wywołanej na rzecz obiektów umieszczonych w tablicy *figury* w klasie *Figury* w metodzie *polozenie*– nie została ona zdefiniowana w tych klasach, a jest wywoływana?



```
C:\Program Files\Xinox Software\JCreatorV3LE\GE2001.exe
false, ze jestem kwadratem, bo jestem Punkt@ad3ba4 ,X=2 , Y=2 , odleglosc=0.0
true, ze jestem kwadratem, bo jestem Kwadrat@1bab50a ,X=2 , Y=2 , odleglosc=2.8284271247461903
false, ze jestem kwadratem, bo jestem Punkt@c3c749 ,X=2 , Y=2 , odleglosc=0.0
true, ze jestem kwadratem, bo jestem Kwadrat@150bd4d ,X=2 , Y=2 , odleglosc=2.8284271247461903
false, ze jestem kwadratem, bo jestem Punkt@1bc4459 ,X=2 , Y=2 , odleglosc=0.0
Press any key to continue..._
```

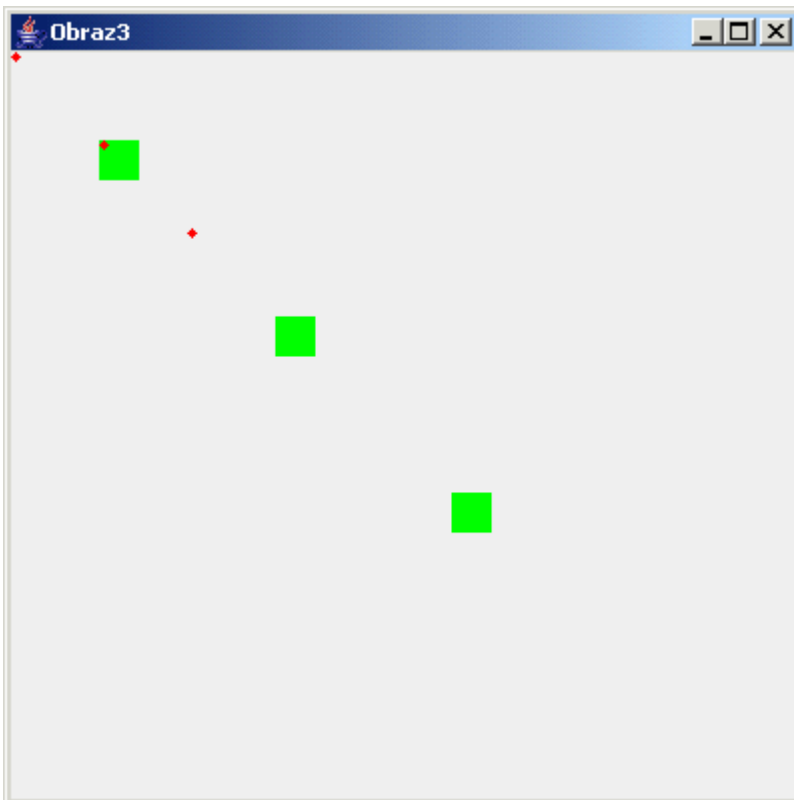
2. Wykonaj program (klasa *Obraz1\_2*), który wyświetla na ekranie dane figur typu *Punkt*, *Kwadrat* tak, jak program *Obraz2* (projekt *Rys2*) oraz dodatkowo dane prostokąta. Klasa *Prostokat* powinna dziedziczyć od klasy *Kwadrat* (definicja konstruktora w klasie *Prostokat*) i dodatkowo powinna posiadać metody obsługujące drugi bok prostokąta (*szerokosc*) oraz metodę *pole* oraz *info*, która przesyła metodę *info* z klasy *Kwadrat*. Klasa *Kwadrat* przesyła metodę *info* z klasy *Punkt*. Klasa *Punkt* musi implementować metodę *info* interfejsu *Wlasciwosci*. W programie *Obraz1\_2*, który ma teraz dodatkowo definicję klasy *Prostokat* w porównaniu z programem *Obraz2*, należy w klasie *Figury* w metodzie *wypelnij()* dodać wstawianie obiektów typu *Prostokat* do tablicy *figury* i uzupełnić metodę *informacje()*.



```
C:\Program Files\Xinox Software\JCreatorV3LE\GE2001.exe
Figura: Punkt, Pole: 0.0
Figura: Kwadrat, Pole: 4.0
Figura: Punkt, Pole: 0.0
Figura: Kwadrat, Pole: 4.0
Figura: Punkt, Pole: 0.0
false, ze jestem kwadratem, bo jestem Punkt@1bab50a ,X=2 , Y=2 , odleglosc=0.0
true, ze jestem kwadratem, bo jestem Kwadrat@c3c749 ,X=2 , Y=2 , odleglosc=2.8284271247461903
false, ze jestem kwadratem, bo jestem Punkt@150bd4d ,X=2 , Y=2 , odleglosc=0.0
true, ze jestem kwadratem, bo jestem Kwadrat@1bc4459 ,X=2 , Y=2 , odleglosc=2.8284271247461903
false, ze jestem kwadratem, bo jestem Punkt@12b6651 ,X=2 , Y=2 , odleglosc=0.0
Press any key to continue...
```

3. Wykonaj program (klasa *Obraz1\_3*), który wyświetla na ekranie dane figur typu *Punkt*, *Kwadrat* tak, jak program *Obraz3* (projekt *Rys3*) oraz dodatkowo dane prostokąta. Klasa *Prostokat* powinna dziedziczyć od klasy *Kwadrat* (definicja konstruktora w klasie *Prostokat*) i dodatkowo powinna posiadać metody obsługujące drugi bok prostokąta (*szerokosc*) oraz metodę *rysuj()* prostokąt w trybie graficznym np. w kolorze niebieskim. W programie *Obraz1\_3*, który ma teraz dodatkowo definicję klasy *Prostokat* w porównaniu z programem *Obraz3*, należy w klasie *Figury* w metodzie *wypelnij()* dodać wstawianie obiektów typu *Prostokat* do tablicy *figury*.

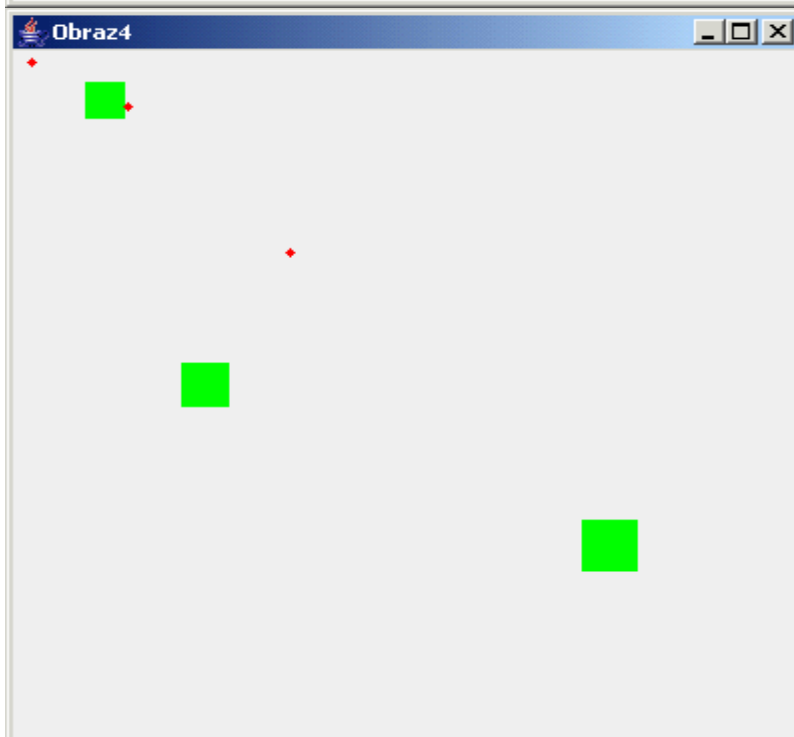
```
C:\Program Files\Xinox Software\JCreatorV3LE\GE2001.exe
false, ze jestem kwadratem, bo jestem Punkt@152c4d9 ,X=0 , Y=0 , odleglosc=62.22539674441618
true, ze jestem kwadratem, bo jestem Kwadrat@1808199 ,X=44 , Y=44 , odleglosc=62.22539674441618
false, ze jestem kwadratem, bo jestem Punkt@1bc887b ,X=44 , Y=44 , odleglosc=124.45079348883236
true, ze jestem kwadratem, bo jestem Kwadrat@a46701 ,X=132 , Y=132 , odleglosc=248.90158697766472
false, ze jestem kwadratem, bo jestem Punkt@166a22b ,X=88 , Y=88 , odleglosc=186.67619023324855
```



4. Wykonaj program (klasa *Obraz1\_4*), który wyświetla na ekranie dane figur typu *Punkt*, *Kwadrat* tak, jak program *Obraz4* (projekt *Rys4*) oraz dodatkowo dane prostokąta i tworzy rysunek animowany. Klasa *Prostokat* powinna dziedziczyć od klasy *Kwadrat* (definicja konstruktora w klasie *Prostokat*) i dodatkowo powinna posiadać metody obsługujące drugi bok prostokąta (*szerokosc*) oraz metodę *rysuj()*, oraz *przesun()* w trybie graficznym. Metoda *przesun* powinna dziedziczyć od klasy *Kwadrat* oraz powinna zmieniać rozmiary boku *szerokosc*. W programie *Obraz1\_4*, który ma teraz dodatkowo definicję klasy *Prostokat* w porównaniu z programem *Obraz4*, należy w klasie *Figury* w metodzie *wypelnij()* dodać wstawianie obiektów typu *Prostokat* do tablicy *figury*.

Animacja jest wykonywana za pomocą metody *animacja* w klasie *Obraz4*, która wywołuje metodę *animacja* z klasy *Figury* przekazując jej rozmiary okna aplikacji *JFrame* (pobiera je za pomocą *getBounds*). Następnie klasa *Obraz4* wprowadza opóźnienie za pomocą metody *Thread.sleep(500)* na 500 milisekund i potem wywołuje metodę *repaint*, która rysuje figury w nowym położeniu – metoda ta wywołuje metodę *paintComponent*. Metoda *animacja* z klasy *Figury* losowo zmienia położenie każdej z figur ich metodami *przesun* podając im wylosowane wartości zmiany współrzędnych *dx* i *dy* oraz bieżące rozmiary okna aplikacji (metoda *przesun* w klasie *Punkt* zmienia wartości współrzędnych *x,y* oraz dba o to, aby były zawarte w granicach bieżącego okna; metoda *przesun* w klasie *Kwadrat* wywołuje metodę *przesun* z klasy *Punkt* oraz zmienia rozmiary swojego boku *dlugosc* w zadanych granicach; metoda *przesun* w klasie *Prostokat* powinna wywoływać metodę *przesun* z klasy *Kwadrat* i dodatkowo zmieniać w zadanych granicach rozmiary boku *szerokosc*).

```
C:\Program Files\Xinox Software\JCreatorV3LE\GE2001.exe
false, ze jestem kwadratem, bo jestem Punkt@152c4d9 ,X=0 , Y=0 , odleglosc=62.22
539674441618
true, ze jestem kwadratem, bo jestem Kwadrat@1808199 ,X=44 , Y=44 , odleglosc=62
.22539674441618
false, ze jestem kwadratem, bo jestem Punkt@1bc887b ,X=44 , Y=44 , odleglosc=124
.45079348883236
true, ze jestem kwadratem, bo jestem Kwadrat@a46701 ,X=132 , Y=132 , odleglosc=2
48.90158697766472
false, ze jestem kwadratem, bo jestem Punkt@166a22b ,X=88 , Y=88 , odleglosc=186
.67619023324855
```



5\*. W zadaniu tym należy do animacji z zadania 4 wprowadzić rozpoznanie zderzania się figur za pomocą metody odległości i np. po wykryciu zderzenia zmieniać ich kolor lub kształt.