

Programowanie obiektowe II

Laboratorium nr 3

Stefan Sokołowski

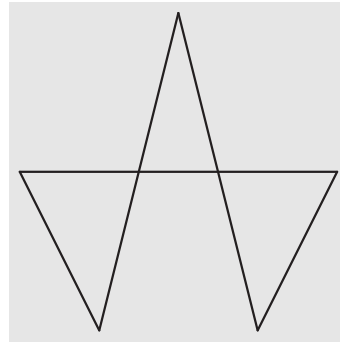
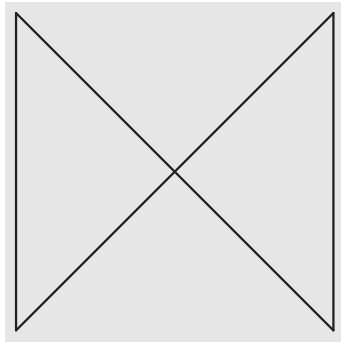
Zadanie 1:

W programie

`iis.ans-elblag.pl/~stefan/Dydaktyka/ProgObiIII/Prog03/P0AbstrWiel.java`

zdefiniowana jest abstrakcyjna klasa `Wielokat` oraz jej konkretne rozszerzenia `Trojkat` i `Czworokat`¹. Proszę dopisać do tego programu takie rozszerzenie klasy `Wielokat`, które będzie w podobny sposób „obsługiwało” pięciokąty. Oczywiście również program główny należy poprawić tak, żeby działał też dla pięciokątów.

Zastosować jakieś sensowne rozwiązanie dla czworokątów i pięciokątów z *samoprzecięciami* obwodów, np. takich:



Zadanie 2:

Na wykładzie² zostało wyjaśnione, jak zdefiniować całkę — funkcję, która jako argument bierze funkcję na liczbach rzeczywistych i wylicza przybliżone pole pod jej wykresem. Przy tym funkcja na liczbach rzeczywistych jest reprezentowana przy pomocy *klasy abstrakcyjnej* albo *interface'u*.

Na wzór całki zaprogramować funkcję, która jako argument bierze funkcję na liczbach rzeczywistych i rysuje gwiazdkami jej przybliżony wykres (z osią x wskazującą w dół a osią y w prawo). Dla uproszczenia możemy się zgodzić, żeby wartości ujemne nie były rysowane, lub były zaokrąglane do zera.

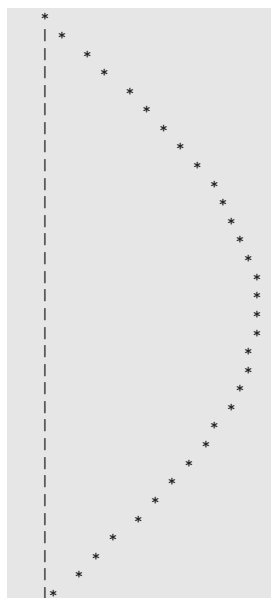
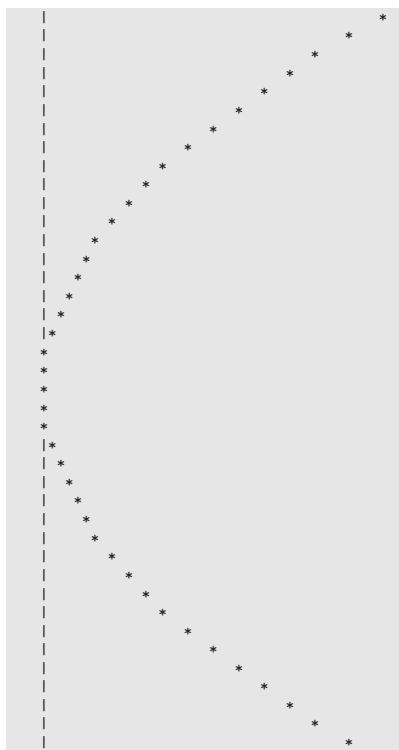
Poniżej są podane dwa przykładowe wykresy funkcji, zdefiniowanych w programie na całkę:

- po lewej wykres funkcji kwadratowej `Kwad`, gdzie $x \in [-2.0 \dots 2.0]$ z krokiem 0.1, i z krokiem 0.1 po osi y ;
- w środku wykres funkcji sinus `Sin`, gdzie $x \in [0.0 \dots \pi]$ z krokiem 0.1, i z krokiem 0.06 po osi y .
- po prawej wykres funkcji sinus `Sin(3*x)`, gdzie $x \in [-2.0 \dots 2.0]$ z krokiem 0.1, i z krokiem 0.08 po osi y .

¹Zgodnie z sugestiami z wykładu 3; por.

`iis.ans-elblag.pl/~stefan/Dydaktyka/ProgObiIII/Slajdy/03a.pdf`, str. 1–4.

²Patrz `iis.ans-elblag.pl/~stefan/Dydaktyka/ProgObiIII/Slajdy/03a.pdf`, str. 7–10.



Oczywiście program powinien rysować wykresy dowolnych funkcji, nie tylko tych trzech.

Uwaga:

W poniższych zadaniach dotyczących sortowania proszę nie korzystać z bibliotecznych metod sortujących Jawy. Proszę programować sortowanie samemu.

Zadanie 3:

Sortowanie bąbelkowe składa się z serii przebiegów tablicy; przebiegi są coraz krótsze, bo nie ma sensu sortować elementów, które już są posortowane. W wyniku pojedynczego przebiegu tablicy największy jej element łąduje na jej końcu, więc następny przebieg może być już o 1 krótszy. Jest to zilustrowane na rysunku z Wikipedii³.

Proszę napisać program sortujący bąbelkowo **właśnie w ten sposób** wczytany ciąg liczb całkowitych (najpierw jego długość, potem wyrazy). Proszę do programu dołączyć licznik, który sprawdzi, ile razy zostało wykonane porównanie wielkości sortowanych liczb całkowitych.

Do testowania można użyć danych z plików

`iis.ans-elblog.pl/~stefan/Dydaktyka/Prog0biII/Prog03/dane-liczby-10tys`

oraz

`iis.ans-elblog.pl/~stefan/Dydaktyka/Prog0biII/Prog03/dane-liczby-100tys .`

Zadanie 4:

Jeśli w którymś przebiegu sortowania bąbelkowego począwszy od pewnego miejsca nie wystąpiła już żadna zamiana, to następny przebieg można skrócić bardziej niż o 1:

³https://en.wikipedia.org/wiki/Bubble_sort#Performance

od miejsca ostatniej zamiany **aż do** końca tablicy
wszystko jest już posortowane i większe od elementów poprzedzających.

Proszę poprawić program z zad. 3, biorąc pod uwagę tą okoliczność.

Zadanie 5:

W zadaniach 3 i 4 chodziło o sortowanie liczb całkowitych. Ale dokładnie tą samą metodę sortowania można wykorzystać do sortowania tablicy dowolnych rzeczy, dla których dysponujemy

- umiejętnością porównywania dwóch elementów: który większy a który mniejszy;
- umiejętnością ich zamieniania w tablicy.

Proszę przerobić program z zad. 4 na sortujący tablicę słów ze względu na porządek alfabetyczny. Do testowania można użyć danych z plików

`iis.ans-elblag.pl/~stefan/Dydaktyka/Prog0biII/Prog03/dane-slowa-10tys`

oraz

`iis.ans-elblag.pl/~stefan/Dydaktyka/Prog0biII/Prog03/dane-slowa-100tys .`

Zadanie 6 (domowe na punkty):

Proszę zdefiniować klasę abstrakcyjną `BubbleSort` zawierającą mechanizm sortowania napisów; korzystając z abstrakcyjnej metody porównywania napisów. Następnie proszę zaprogramować

- rozszerzenie klasy `BubbleSort` o porównanie alfabetyczne napisów;
- rozszerzenie klasy `BubbleSort` o odwrotne porównanie alfabetyczne napisów; czyli $napis_1$ poprzedza $napis_2$ jeśli $napis_2$ jest alfabetycznie wcześniejszy od $napis_1$;
- rozszerzenie klasy `BubbleSort` o porównanie alfabetyczne *a tergo* (od końca); czyli o pierwszeństwie decyduje ostatnia litera; a jeśli są równe, to przedostatnia; itd.

Uwaga:

Czas na przysyłanie (na `s.sokolowski@ans-elblag.pl`): 2 tygodnie od zajęć z lab. 3 — potem już nie przyjmuję rozwiązań. Liczba punktów zależy od oryginalności i elegancji rozwiązania.

Zadanie 7:

Proszę napisać

- (a) metodę biorącą jako argument tablicę liczb rzeczywistych i oddającą jako wartość **średnią arytmetyczną** tych liczb:

$$\frac{a_0 + a_1 + a_2 + \dots + a_{n-1}}{n}$$

- (b) metodę biorącą jako argument tablicę liczb rzeczywistych i oddającą jako wartość **średnią geometryczną** tych liczb:

$$\sqrt[n]{a_0 \cdot a_1 \cdot a_2 \cdot \dots \cdot a_{n-1}}$$

- (c) metodę biorącą jako argument tablicę liczb rzeczywistych i oddającą jako wartość **średnią harmoniczną** tych liczb:

$$\frac{n}{\frac{1}{a_0} + \frac{1}{a_1} + \frac{1}{a_2} + \dots + \frac{1}{a_{n-1}}}$$

- (d) metodę biorącą jako argument tablicę liczb rzeczywistych i oddającą jako wartość **maksimum** tych liczb:

$$\max(a_0, a_1, a_2, \dots, a_{n-1})$$

Napisać program wczytujący ciąg liczb rzeczywistych, stosujący te 4 metody i drukujący wyniki.

Zadanie 8 (domowe specjalne):

Następujące operacje potrzebne są do policzenia średniej arytmetycznej danego ciągu liczb rzeczywistych:

- element neutralny dodawania (zero), żeby mieć od czego zacząć sumowanie;
- dodawanie liczb rzeczywistych;
- dzielenie liczby rzeczywistej przez całkowitą (długość ciągu).

W pliku

`iis.ans-elblag.pl/~stefan/Dydaktyka/ProgObiIII/Prog03/P6.SredAbstr.java`

zdefiniowana jest klasa abstrakcyjna `Srednia` obliczająca średnią przy użyciu takich trzech operacji; a następnie zdefiniowane jest jej rozszerzenie `Arytm` definiujące te operacje tak, żeby jej metoda `srednia` dawała średnią arytmetyczną. Proszę sprawdzić, że to działa.

Tą samą klasę można wykorzystać do liczenia innych rzeczy niż średnią arytmetyczną, wystarczy inaczej podeszyc do definiowania operacji podstawowych. Proszę więc do tego programu dodać następujące rozszerzenia klasy `Srednia`:

- `Geom` dla liczenia średniej geometrycznej,
- `Harm` dla liczenia średniej harmoniczej,
- `Max` dla liczenia maksimum

(por. zad. 7). Każde z tych rozszerzeń powinno tylko zdefiniować te metody, które w klasie `Srednia` są oznaczone jako abstrakcyjne.

Następnie proszę poprawić wydruk w klasie głównej tak, żeby dla wczytanego ciągu liczb rzeczywistych program drukował wyniki wszystkich czterech operacji.

Uwaga:

Pierwszej osobie, która przyśle mi (na `s.sokolowski@ans-elblag.pl`) poprawne rozwiązanie tego zadania, przysługują punkty do zaliczenia laboratorium. Liczba tych punktów zależy od oryginalności i elegancji rozwiązania.
