



Polsemestrálny test

Zadanie



Ústav informatiky
Prírodovedecká fakulta
UPJS v Košiciach

Dvakrát meraj (rozmyšľaj), raz rež (programuj)

Pravidlá a informácie:

- o čas na riešenie úloh je **80 minút**, resp. do 11:20,
- o nie je dovolená žiadna (elektronická ani neelektronická) komunikácia s kýmkoľvek okrem dozoru,
- o nie je dovolené používať žiadne zdroje ani materiály okrem oficiálneho ťaháku a predmetového webu vrátane gitlabu,
- o nie je dovolené používať žiadnu inú aplikáciu než Eclipse (s výnimkou webového prehliadača pri odosielaní riešenia),
- o porušenie pravidiel má za následok hodnotenie FX,
- o svoje riešenia odovzdávajte cez systém Moodle (<http://lms.ics.upjs.sk/>).

Ktoré úlohy treba riešiť:

V **Časti 1** je cieľom úloh vytvoriť triedu Midtermarka, ktorá rozširuje triedu Turtle. Z prvej trojice úloh si **vyberte len 2 úlohy**, ktoré **budete riešiť!!!** To, ktoré úlohy ste sa rozhodli riešiť, uveďte v komentári pri odosielaní riešenia cez Moodle (ak to nie je zrejmé z odoslaného).

V **Časti 2** je len jedna úloha, t.j. v tejto časti nie je možný výber úloh.

Časť 1 (dve úlohy z troch)

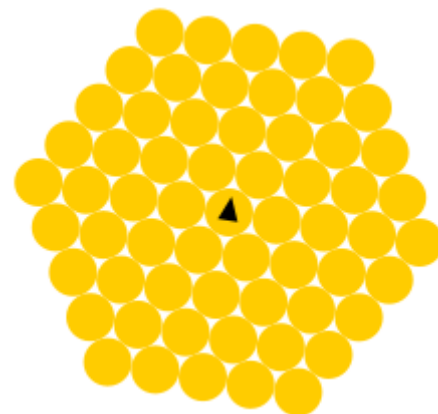
Dotted hexagon (10 bodov)

Do triedy Midtermarka pridajte metódu dottedHexagon, ktorá nakreslí vzor ako na obrázku. Metóda má dva parametre `sideLength`, `radius`. Vzor je tvorený kruhmi. Kruhy majú polomer určený parametrom `radius` a ich farba má RGB zložky 255, 204, 0. Parameter `sideLength` určuje počet kruhov od stredu po okraj. Prvý obrázok má parametre 5, 12, druhý obrázok má parametre 1, 10, tretí obrázok má parametre 2, 10 a štvrtý 3, 8. Korytnačka sa na začiatku aj na konci nachádza v strede obrazca. Obrázec je nakreslený v smere natočenia korytnačky. Po vykonaní metódy nech je korytnačka na pozícii a v smere, ako bola pred vykonaním metódy.

```
public void dottedHexagon(int sideLength, double radius)
```

Rada:

Odporúčame si vytvoriť pomocnú metódu, ktorá nakreslí jeden riadok.



Na druhej strane papiera nájdete oficiálny ťahák.

Friendly pair of numbers (10 bodov)

Do triedy `Midtermarka` pridajte metódu `friendlyPair`. Táto metóda dostane dva parametre kladné celé čísla m a n . Označme $\sigma(n)$ súčet deliteľov čísla n . Dvojicu čísel nazveme priateľskou ak platí rovnosť $\sigma(n)/n = \sigma(m)/m$. Príklady:

`friendlyPair(30, 140) = true`, lebo dostávame $(1+2+3+5+6+10+15+30)/30 = 72/30 = 12/5$
 $(1+2+4+5+7+10+14+20+28+35+70+140)/140 = 336/140 = 12/5$,

`friendlyPair(6, 28) = true`, lebo dostávame $(1+2+3+6)/6 = 12/6 = 2/1$
 $(1+2+4+7+14+28)/28 = 56/28 = 2/1$,

`friendlyPair(8, 28) = false`, lebo dostávame $(1+2+4+8)/8 = 15/8$
 $(1+2+4+7+14+28)/28 = 56/28 = 2/1$.

```
public boolean friendlyPair(int m, int n)
```

Najdlhšie číslo (10 bodov)

Do triedy `Midtermarka` pridajte metódu `digitsInLongestNumber`. Táto metóda dostane ako parametre `null`ovú referenciu na reťazce (objekty triedy `String`) a vráti hodnotu, ktorá vyjadruje počet cifier v najdlhšom celom čísle, ktoré sa nachádza v reťazci.

Príklady (najdlhšie číslo je zobrazené **hrubým**):

```
digitsInLongestNumber("Od 21.12.2007 je Slovensko členom  
Schengenského priestoru.") = 4,
```

```
digitsInLongestNumber("193 12 58967 12") = 5,
```

```
digitsInLongestNumber("5+5=10") = 2,
```

```
digitsInLongestNumber("Akademicky rok 2023/2024") = 4,
```

```
digitsInLongestNumber("James Bond 007") = 3,
```

```
digitsInLongestNumber("Milan je plavčík") = 0.
```

Pozn.: číslom myslíme súvislý podreťazec pozostávajúci iba z cifier.

```
public int digitsInLongestNumber(String s)
```

Časť 2

Korytnačky v ploche (10 bodov)

- (3 body) Vytvorte triedu `MidtermPane`, ktorá rozširuje triedu `WinPane`. Po vytvorení kresliacej plochy triedy `MidtermPane` nech sa v nej (automaticky v konštruktore, resp. „inicializačnej metóde“) vytvorí 7 korytnačiek triedy `Turtle`. Korytnačky sú na náhodných pozíciách vo viditeľnej časti kresliacej plochy, pričom x aj y súradnica je deliteľná číslom 50 bez zvyšku.

Pozn.: Ak riešite iba túto časť tak pridajte prázdnu metódu zo 7 bodovej úlohy kvôli evaluácii.

- (7 bodov) Do triedy `MidtermPane` pridajte metódu `coverByRectangle`. Táto metóda nakreslí najmenší obdĺžnik, taký aby každá korytnačka v ploche sa nachádzala v ňom alebo na jeho okraji. Zároveň sú strany obdĺžnika rovnobežné s okrajmi plochy. Metóda vráti obvod tohoto obdĺžnika.

```
public int coverByRectangle()
```

