

Dokumentáció

I. Beadandó Programozási technológia

Készítette:
Gönye Mihály

Neptun kód:
PZNY6

Dátum:
2023. 09. 30.

Tartalom

1.	A feladat leírása	2
2.	A terv.....	2
2.1.	Típusok.....	2
2.1.1.	Point	2
2.1.2.	Shape	2
2.1.3.	RegularPolygon	2
2.1.4.	Konkrét síkidomok	3
2.1.5.	Kivételek kezelése	3
2.1.6.	Shapes	3
2.2.	Osztálydiagram	4
2.3.	Implementálás	4
3.	Tesztelés.....	5
3.1.	Fehérdobozos tesztesetek.....	5
3.2.	Feketedobozos tesztesetek	5

1. A feladat leírása

Rögzítsen a síkon egy pontot, és töltsön fel egy gyűjteményt különféle szabályos (kör, szabályosháromszög, négyzet, szabályos hatszög) síkidomokkal! **Számolja meg, hogy a pontot hány síkidom tartalmazza!** Minden síkidom reprezentálható a középpontjával és az oldalhosszal, illetve a sugárral, ha feltesszük, hogy a sokszögek esetében az egyik oldal párhuzamos a koordináta rendszer vízszintes tengelyével, és a többi csúcs ezen oldalra fektetett egyenes felett helyezkedik el. A síkidomokat szövegfájlból töltse be! A fájl első sorában szerepeljen a síkidomok száma, majd az egyes síkidomok. Az első jel azonosítja a síkidom fajtáját, amit követnek a középpont koordinátái és a szükséges hosszúság. A feladatokban a beolvasáson kívül a síkidomokat egységesen kezelje, ennek érdekében a síkidomokat leíró osztályokat egy közös ősosztályból származtassa!

2. A terv

2.1. Típusok

2.1.1. Point

A síkidomokat derékszögű koordinátarendszerben ábrázoljuk. A síkidomok középpontjának reprezentálásához, valamint csúcsai koordinátáinak meghatározásához szükségünk lesz a `Point` osztályra, mely két adattaggal (`double x, y`) rendelkezik. Ez az osztály csak egy konstruktorral rendelkezik, melynek segítségével az adattagok értékeit állíthatjuk be.

2.1.2. Shape

A `Shape` egy absztrakt osztály, melyből a különböző síkidomokat származtatjuk. Egyetlen adattagot tartalmaz, mégpedig a síkidom középpontját (`Point center`). Itt kerül deklarálásra a `isInTheShape()` metódus. A feladatban szereplő síkidomokat két csoportba osztjuk:

1. Kör
2. Szabályos sokszög (például szabályos háromszög, négyzet, szabályos hatszög)

2.1.3. RegularPolygon

A szabályos sokszögeket az absztrakt `RegularPolygon` (szabályos sokszög) osztályból származtatjuk, mely a `Shape` osztályból származtatott. Itt vezetjük be az `side` (oldalhossz) adattagot és az `isAboveOrOnTheLine()` metódust, melyet később arra használunk, hogy eldöntsük azt, hogy egy pont egy egyenes fölött helyezkedik el, vagy sem.

2.1.4. Konkrét síkidomok

A `Circle` (Kör) osztályt a síkidomok osztályából származtatjuk, majd felvesszünk neki egy `radius` (sugár) nevű adattagot anélkül, hogy legyen egy feleslegesen megörökölt oldalhossz adattagunk.

A többi szabályos síkidomot (`EquilateralTriangle`, `Square`, `RegularHexagon`) a `RegularPolygon` osztályból származtatjuk.

Az utóbbi osztályokon belül van definiálva az `isInTheShape()` metódus, mely igazgal tér vissza, ha benne, illetve hamissal ha nincs benne a megadott pont.

A bemeneti fájlokban a következőképpen jelöljük az alábbi síkidomokat.

Jelölés	Síkidom
circle	kör
square	négyzet
triangle	szabályos háromszög
hexagon	szabályos hatszög

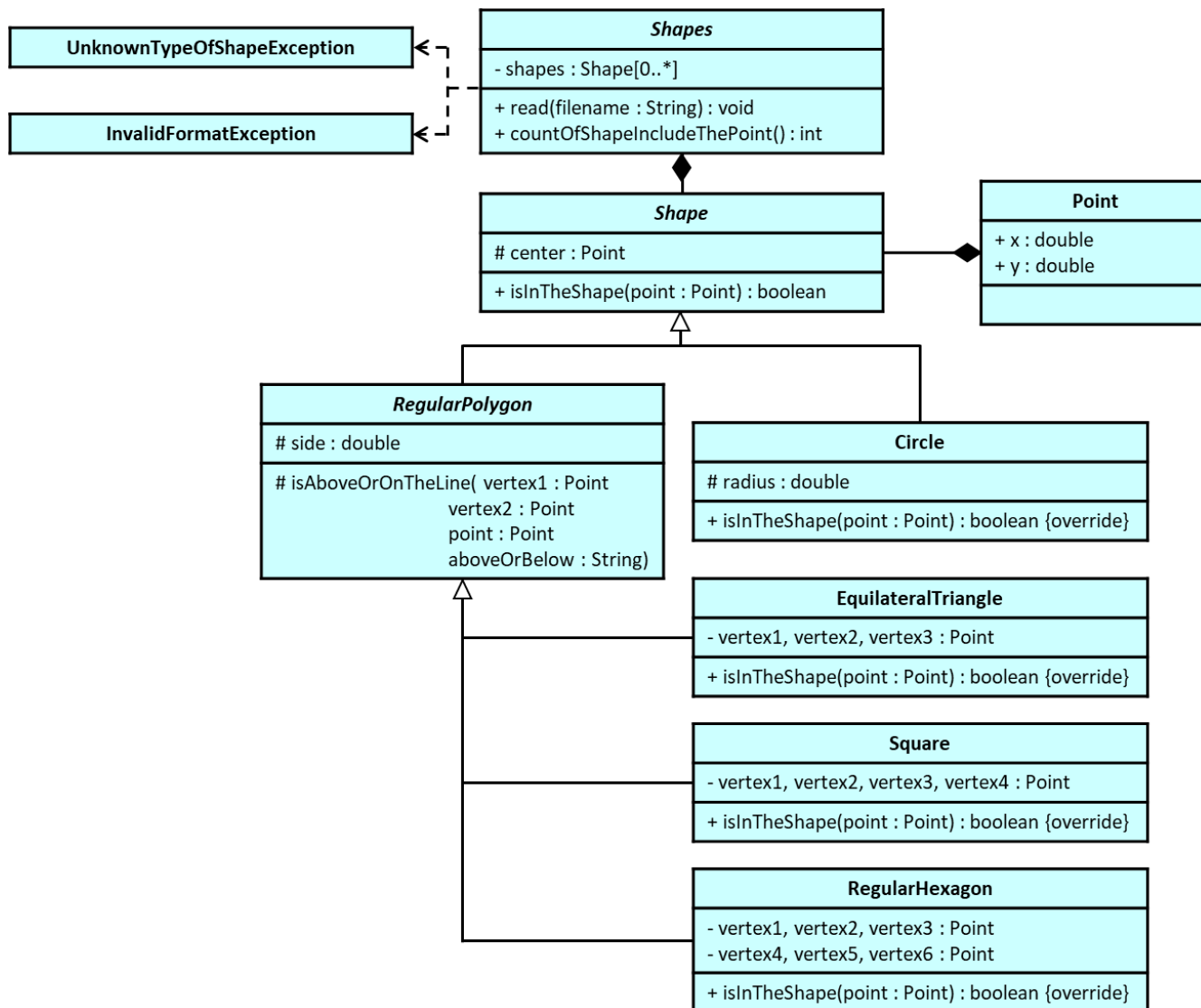
2.1.5. Kivételek kezelése

Bevezetjük az `InvalidFormatException` és az `UnknownShapeTypeException` kivételeket, melyek akkor lépnek fel, ha nem megfelelő a bemeneti fájlban szereplő adatok formátuma, vagy pedig, ha olyan jelölést tartalmaz a fájl, mely nem határoz meg egy síkidomot sem.

2.1.6. Shapes

A `Shape` osztályon belül létrehozunk egy síkidomokat tároló adatszerkezetet, valamint itt definiáljuk a beolvasást végrehajtó `readShapes()` függvényt. Itt található a `countOfShapeIncludeThePoint()` metódus, mely a feladat kérdésére válaszolja meg.

2.2. Osztálydiagram



2.3. Implementálás

A **Point** osztállyal érdemes kezdeni az implementálást, mivel ez az osztály nem függ semmitől. majd folytassuk az általánosabb osztályoktól a specifikusabbak felé.

Használt Java verzió: 19.0.2

3. Tesztelés

3.1. Fehérdobozos tesztesetek

Leírás	Bemenet	Elvárt kimenet
Nem létező fájl	shapesFileNotFound.txt	FileNotFoundException
Üres fájl	shapesEmptyFile.txt	InvalidFormatException
Hibás formátumú fájl #1	shapesWrongFile1.txt	InvalidFormatException
Hibás formátumú fájl #2	shapesWrongFile2.txt	InvalidFormatException
Ismeretlen síkidom	shapesUnknownTypeOfShape.txt	UnknownShapeTypeException
Helytelen argumentum	shapesIllegalArgument.txt	IllegalArgumentException

3.2. Feketedobozos tesztesetek

Leírás	Bemenet	Elvárt kimenet
Nulla síkidom	shapes0.txt	The point is contained by 0 shapes.
Egy síkidom - a síkidom tartalmazza	shapes1.txt	The point is contained by 1 shapes.
Több síkidom - 2 síkidom tartalmazza	shapes2.txt	The point is contained by 2 shapes.

