

Tóth Botond

1.beadandó/10.feladat

2021.08.02.

MQH41V

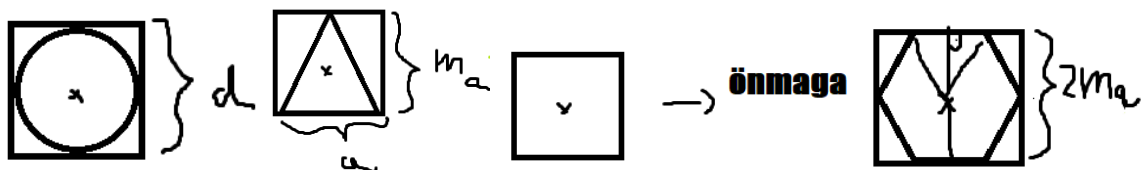
[mqh41v@inf.elte.hu](mailto:mqh41v@inf.elte.hu) vagy [tothbotond00@gmail.com](mailto:tothbotond00@gmail.com)

## Feladat

Töltsön fel egy gyűjteményt különféle szabályos (kör, szabályos háromszög, négyzet, szabályos hatszög) síkidomokkal! Adja meg melyik síkidom befoglaló téglalapja a legnagyobb területű! Egy síkidom befoglaló téglalapja lefedi a síkidomot, oldalai párhuzamosak a tengelyekkel. 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 őssztályból származtassa!

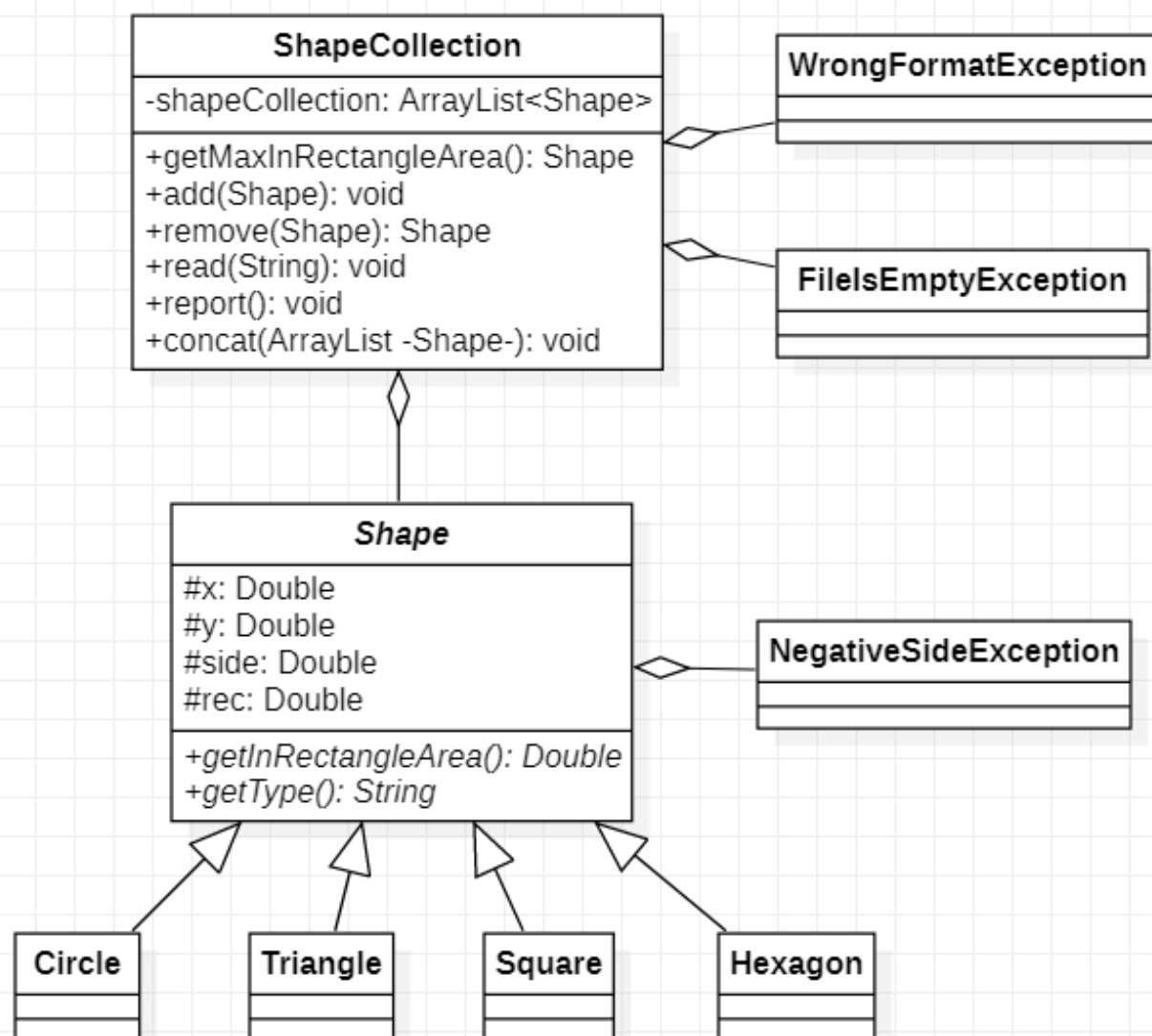
## Megoldás terve

Először érdemes vizualizálni a feladatot, hogy kiderüljenek a szükséges számítások:



A befoglaló téglalapok területének kiszámításához a megoldásban szinte csak az oldal hosszakra lesz szükségünk. Legyen egy absztrakt alakzat osztályunk majd ebből származtathatjuk a konkrét típusokat. Legyen egy fájl ami ilyen alakzatok neveit és adatait tartalmazza és ennek beolvasására készítsünk egy külön metódust. Legyen még egy osztályunk amiben ezeket az alakzatokat eltároljuk egy gyűjteményben (pl. ArrayList) majd legyen lehetőségünk ebből lekérdezni a legnagyobb területű befoglaló téglalapot ( maximum kiválasztás). Az osztályhoz lehessen hozzáadni alakzatokat is! Hibakezelést custom Exceptionokkal oldjuk meg!

## Osztálydiagramm



## Osztályleírások

### Shape:

Az alakzatok absztrakt ősosztálya. Belőle származnak a konkrét alakzattípusok: a Circle, a Triangle, a Square és a Hexagon. A Shape osztály lényegében az egész program alaköve és ennek adataival lehet kiszámítani a befoglaló téglalapok területeit. NegativeSideException-t dob a konstruktor ha negatív oldalhosszat akarunk megadni.

### Circle, Triangle, Square, Hexagon:

A Shape osztályból származtatjuk őket. Megvalósítják a Shape absztrakt metódusait. Mindegyik osztály a maga módján számolja ki a befoglaló téglalapja területét.

### ShapeCollection:

Az alakzatokat tartalmazó gyűjtemény. A Shape-eket egy ArrayList-ben tárolja el. Ehhez a listához kedvünkre adhatunk hozzá és vehetünk ki Shape-eket. A legfontosabb metódusa a getMaxInRectangle() mely megadja azt az Shape-t aminek a legnagyobb befoglaló téglalapja van. Ilyen tömböt akár már létező tömbből is létre tudunk hozni. Az osztály emellett képes fájlból beolvasni és az abban található valid alakzatokat a listába beszúrni.

Megoldás: a ShapeCollection report metódusával jeleníthető meg.

## Tesztelési terv

### Fekete doboz:

Input File neve	Teszt	Bemenet	Várt kimenet
test1.txt	0 méretű tömb	üres file	valamilyen error
test2.txt	1 méretű tömb	1 sort tartalmazó file	visszadobja a beadott adatot
test3.txt	több méretű tömb	10 sort tartalmazó file	tényleges maximum kiválasztása
test4.txt	több méretű tömb	5 valid és 5 invalid sort tartalmazó file	maximum kiválasztás csak a jó adatokkal
test5.txt	több méretű tömb	két adat is maximum lehet	hamarabbi indexű maximummal tér vissza
test6.txt	0 méretű tömb	file teli rossz adatokkal	valamilyen error
-	nem létező file	nincs	valamilyen error

### Fehér doboz:

Input File neve	Teszt	Bemenet	Várt kimenet
test7.txt	kör a maximum	file kör maximummal	a kör
test8.txt	négyzet a maximum	file négyzet maximummal	a négyzet
test9.txt	hatszög a maximum	file hatszög maximummal	a hatszög
test10.txt	háromszög a maximum	file háromszög maximummal	a háromszög
test11.txt	terhelés	extra sok bemenet	maximum megtalálása