

Harmadik programozás házi feladat

Math Function Visualizer: Specifikáció

Adatok:

- Név: Morvai Barnabás
- Neptun: E0HB0N
- Státusz: Specifikáció
- Dátum: 2024.10.22.

Szöveges leírás

Ablakban megnyitható programot szeretnék írni, matematikai függvények koordináta-rendszeren való megjelenítésére, szerkesztésére és azok éppen aktuális állapotának mentésére.

Tervezett UI/UX

A függvények kirajzolása egy testreszabható koordináta-rendszeren fog történni. A felhasználó olyan állapot tud létrehozni amilyet a feladat megkíván vagy szeretne. Ebbe beletartozik az azt alkotó vonalak színe, szélessége, az egyes beosztások közötti távolság vagy azok hosszúsága pixelben. Ezen felül a létrehozott koordináta-rendszerben szabadon fog tudni mozogni azáltal, hogy be- vagy kizoomol az egerének a görgőjével.

Függvényt a képernyő jobb oldalán található menüsávban tud hozzáadni vagy elvenni. Minden éppen kirajzolt függvényt testre fog tudni szabni annak minden matematikai formalizmusával, például hatvány- vagy lineárfüggvények attribútumait szabadon változtathatja, átírhatja a program futásának bármely szakaszában. Hozzáadáskor egy előre meghatározott listából tud majd választani, mint például a korábban említett típusok, de természetesen széles skáláról lehet választani, akár trigonometrikus függvények, akár exponenciális. Miután kiválasztotta a felhasználó, hogy melyiket szeretné látni, paraméterezheti azt, mint például eltolás X vagy Y irányban, lineáris esetben, hogy mennyire legyen meredek a görbe, hatványfüggvény esetében pedig, hogy milyen kitevőre emelje majd a változót. Egyes elemek paraméterezhetősége ismertetve lesz a kitöltendő cellák felett. Emellett szabadon változtatni lehet a függvények színét vagy vonaluk vastagságát külön-külön.

Abban az esetben ha a felhasználó be akarja zárni az alkalmazást, de később vissza szeretné nézni, a már definiált görbéit ezt könnyen megteheti majd az aktuális állapot mentésével, amit később szabadon be fog tudni tölteni és azt látja majd viszont, ahol legutóbb abbahagyta.

Hasonló elemi, funkciók lesznek elérhetőek az ablak tetején található menüsávban, mint a mentés, visszatöltés, undo, redo vagy információk a projektről három különböző lenyitható fülön, amik a File, Edit és Help névre hallgatnak majd. Ezeket lenyitva lesznek elérhetőek a fent felsoroltak.

Megoldás ötlete

A függvények ábrázolásáért egy Java-ban megtalálható Polygon osztály fog felelni, egy ilyen Polygon példány úgy fog létrejönni, hogy a hozzá tartozó add metódusban megadom, hogy az adott X tengelyen lévő pontokhoz milyen Y tengelyen található pontok fognak tartozni az adott függvény természetétől függően. Miután az egész, a függvényeknek szánt ablak szélességében meg lett határozva a Polygon értéke egy drawPolyline metódussal kirajzolom a képernyőre.

A különböző típusú függvényeknek lesz egy közös Function ősosztályuk, ahol el lesz tárolva, hogy hány pixel egy beosztás a koordináta-rendszerben a két tengelyen. Ez a Function osztály pedig leszármazottja lesz a JPanel-nek, hogy hozzá lehessen adni őket egy JLayeredPanel-ből származó saját osztályomhoz, hogy aztán ki lehessen őket rajzolni a képernyőre.

Az ablakokat JFrame használatával fogom megoldani, alapértelmezetten egy átlagos FullHD monitorhoz igazítva, tehát 1920x1080 pixel. Alapból be lesz állítva, hogy az X gombbal be lehessen zárni, de a File menüben található Exit-tel is be lehet fejezni a munkamánst egy System.exit(0)-val. A fenti menüsáv egy JMenuBar, amin a leírásban található lenyitható, kattintható JMenu elemeket tartalmaz majd. A kattintható elemek egy ActionListener-t fognak implementálni, ahol definiálva lesz, hogy mi történjen, amikor valamelyiket lenyomja a felhasználó.

Ezeket mindezt úgy fogom implementálni, hogy nem konkrétan az adott osztályokat fogom használni, hanem egy saját hasonló osztályt írok, amik leszármaznak az adott elemből. Mindezt azért, hogy amikor használok őket ne ott keljen beállítani minden tulajdonságukat, hanem ezt már rögtön a konstruktorban elvégezhessem. Így mikor újabb példányt definiálok belőlük rögtön az általam kívánt beállításokkal fognak rendelkezni. Például: *public class Frame extends JFrame*, *public class Menu extends JMenu*, *public class MenuBar extends JMenuBar* stb.