



PROGRAMOZÁS ALAPJAI 2. (VIAUAA01)

3. LABOR IMSC FELADAT

BEVEZETÉS A POLIMORFIZMUSBA

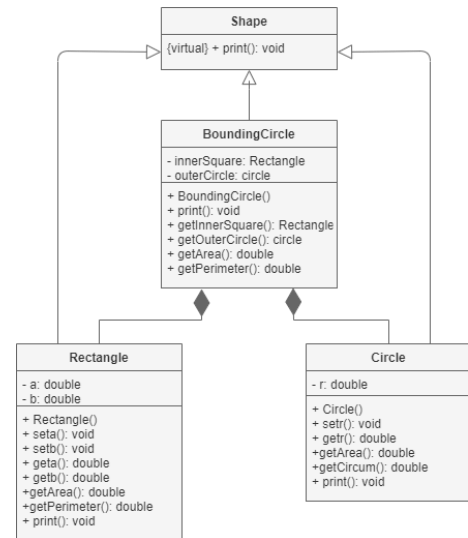
KOVÁCS LEVENTE (F5UHYT)

6. Bevezetés a polimorfizmusba

A felvetett problémát esszenciájában az jelenti, hogy a különböző hasonló felépítésű és működésű *Rectangle*, *Circle*, *Bounding Circle* osztályokra működésükben különböző, de valójában ugyanazt a feladatot elvégző függvényeket kéne írni, meghívni. Jelen esetben ez a véletlenszerű feltöltés és a kiírás. Erre a problémára a megoldást az jelentené, ha a különböző objektumokra tudnánk bevezetni és jól definiálni egy közös interfacet. Vagyis tulajdonképpen egyszerűen csak csoportosítani kéne a hasonló tulajdonságokkal rendelkező objektumokat egy nagyobb objektumhalmazba, majd implementálni a közös tulajdonságok alapján egy összefogó interakciós felületet.

A fentiek alapján a megoldást a polimorfizmus jelenti. Az ilyen összefogó, vagy alárendelt típusokkal rendelkező megvalósítást altípusos polimorfizmusnak hívják. Az ötlet az, hogy több hasonló osztályt foglaljunk egy szélesebb ősosztályba. Jelen esetben például egy ilyen ősosztály lehetne egy *Shapes* nevű class, mely leszármazottjai a *Rectangle*, *Circle*, *Bounding Circle*. Az altípusos polimorfizmus lényege abban rejlik, hogy a különböző osztályokban különbözően megvalósított, de ugyanazt a funkciót ellátó metódusokra (függvényekre) létrehozunk az ősosztályban egy ugyanolyan nevű virtuális metódust. Amennyiben a virtuális (*virtual*) metódust hívjuk meg (például egy *Shape**-on keresztül egy virtuális *print* függvényt), akkor amennyiben a leszármazott típusban is létezik azonos nevű függvény, az felülírja (*override*) az ősosztály függvényét és az fog lefutni.

Így egyszerű a konkrét megvalósítása a fenti problémára adott megoldásnak: Létrehozunk egy *Shapes* osztályt, amelynek leszármazottjai a többi alakzat és definiálunk az ősosztályban egy `virtual void print() const;` metódust. Végül a létrehozunk egy *Shape shapes[300]* tömböt, így amikor a *shapes[i]->print();* függvényhívás történik, akkor egyszerűen a mutatott objektumnak megfelelő leszármazott osztályban definiált függvény fog lefutni és kiírni.



Osztálydiagram polimorfizmussal