

1. Típus és osztály

Határidő márc 5, 23:59

Pont 10

Kérdések 10

Időkorlát Nincs

Engedélyezett próbálkozások 5

[Kvíz kitöltése újra](#)

Próbálkozások naplója

	Próbálkozás	Idő	Eredmény
LEGUTOLSÓ	1. próbálkozás	12 perc	7 az összesen elérhető 10 pontból

⚠ A helyes válaszok el vannak rejtve.

Ezen próbálkozás eredménye: **7** az összesen elérhető 10 pontból

Beadva ekkor: feb 28, 20:27

Ez a próbálkozás ennyi időt vett igénybe: 12 perc

1. kérdés

1 / 1 pont

Mi jellemzi az objektum elvű programozási paradigmát?



Egy probléma megoldáshoz szükséges adatok egy-egy részét a hozzájuk kapcsolódó tevékenységekkel együtt egységekbe foglalja, hogy a megoldás folyamatát ezen egységek tevékenységei közötti vezérlés-átadások határozzák meg.



Egy probléma részfeladatait megoldó tevékenységeket önálló egységekbe szervezi, hogy a megoldás folyamatát ezen tevékenységei közötti vezérlés-átadások határozzák meg.



Egy probléma megoldása érdekében objektumokat példányosít, hogy azok egymás közötti kommunikációja oldja meg a feladatot.



A megoldandó feladat modelljéhez osztály diagrammot használ.

2. kérdés**1 / 1 pont**

Mire szolgál egy feladat specifikációjának az utófeltétele?



Megmutatja a feladat inputváltozóinak kezdőértékeire vonatkozó megszorításokat.



Ez a logikai állítás akkor igaz, ha a feladat megoldására szánt program helyesen működik.



Megadja, hogy a feladat megoldásához a változóknak milyen végértékeket kell felvenni.



Megmutatja, hogyan kell a feladat input adataiból előállítani az output adatokat.

3. kérdés**1 / 1 pont**

Mit jelent a „végrehajtható specifikáció” fogalma?



A megoldandó feladat specifikációjának utófeltételében a nevezetes algoritmus minták szimbólumainak segítségével fogalmazzuk meg a feladat célját.



A feladatnak egy olyan leírása, amelyet közvetlenül futtathatunk a számítógépen.



A specifikáció a feladat leírásán túl megmutatja azt is, hogy milyen módon (milyen algoritmussal) lehetne a feladatot megoldani.



Az elő-, utófeltételes specifikációkat nevezzük így, amennyiben azok input-, outputadatok helyett változók bevezetésével fogalmazzák meg a feladatot.

4. kérdés

1 / 1 pont

Milyen információt ad meg egy adatról (változóról) annak a típusa?



Megmutatja, hogy elemi vagy összetett-e az adat, illetve, hogy érték szerint, vagy referencia szerint kell-e tárolni.



A változó lehetséges értékeinek halmazát, az értékekkel végezhető műveleteket, az értékek reprezentálását, és a műveletek implementációját.



A változó által felvehető lehetséges értékeknek a halmazát, és ezen értékeknek a számítógépen történő ábrázolásából fakadó korlátait.



A változó által felvehető lehetséges értékeknek a halmazát, illetve azt a nevet, amivel azonosíthatjuk a típust.

Helytelen

5. kérdés

0 / 1 pont

Milyen a jó típusreprezentáció?



Amelyik egy vagy több elemi, illetve összetett (például tömb) típusú adattaggal képes helyettesíteni egy típusértékeket úgy, hogy az adattagok kielégítik a típusinvariánst.



Amelyik olyan típusinvariánst kielégítő elemek együttesével helyettesít egy típusértéket, amelyek egy konkrét programozási nyelven közvetlenül megadhatók.



Amelyik minden típusműveletet olyan programmal helyettesít, amelyben a típusértékeket a típusinvariánst kielégítő reprezentánsaik helyettesítik.



Amelyik bármelyik típusértéket olyan elemek együttesével helyettesít, amely kielégíti a típusinvariánst, és a típusinvariánst kielégítő elemek egy együttese egy típusértéket helyettesít.

6. kérdés

1 / 1 pont

Milyen a jó típusimplementáció?



Amelyik minden típusműveletnek megfeleltet egy olyan programot, amelyben a típusértékeket a típusinvariánst kielégítő reprezentánsaik helyettesítik.



Amikor a típusműveleteket helyettesítő programok egy az egyben kódolhatók egy konkrét programozási nyelven, beleértve, hogy a típusértékeket helyettesíthető reprezentáns elemek is leírhatók azon a nyelven.



Amikor a típusműveleteket helyettesítő programok helyreállítják a típusértékeket helyettesítő reprezentáns elemek típusinvariánsát.



Amelyik bármelyik típusértéket olyan elemek együttesével helyettesít, amely kielégíti a típusinvariánst, és a típusinvariánst kielégítő elemek együttese egy típusértéket helyettesít.

7. kérdés

1 / 1 pont

Mit jelent az egységbe zárás fogalma?



Azt, amikor a feladatot megoldó program egyes részeit metódusként a választott programozási nyelv erre a célra szolgáló nyelvi egységben (pl. class) helyezzük el.



Azt, amikor egy adott feladatkör megvalósításához szükséges adatokat és az azokat manipuláló programrészeket a program többi részétől elkülönítve együtt adjuk meg.



Azt, amikor ugyanazon osztályhoz tartozó objektumokat egyazon programblokkban példányosítjuk, ott, ahonnan az osztály definíciója is látható.



Azt, amikor az egységbe zárt adattagok és metódusok egy részének láthatóságát korlátozzuk, hogy azokra az osztályon kívül ne hivatkozassunk.

Helytelen

8. kérdés

0 / 1 pont

Mit értünk elrejtés alatt?



Amikor lokális változókat vezetünk be egy metódusban, és ezeket a program más részei nem láthatják, nem használhatják.



Egy metódus számára rejtett az az objektum, amire a metódust meghívják, hiszen ez a metódus deklarációjában nem látszik: ott csak a paraméterváltozók típusa és a visszatérési érték típusa jelenik meg.



Amikor egy osztály egy metódusa az osztály egyik adattagjának nevével és típusával azonos lokális változót definiál, amely ezáltal elfedi (eltakarja) az adattagot, így azt ez a metódus nem láthatja, nem használhatja.



Amikor az egységbe zárt adattagok és metódusok egy részének láthatóságát korlátozzuk, hogy azokra az osztályon kívül ne hivatkozassunk.

9. kérdés

1 / 1 pont

Melyek az osztálydefiníció részei?



Osztály neve, a belőle példányosítható objektumok adattagjai (nevükkel és típusukkal) és metódusai (nevükkel, paraméterlistájukkal, visszatérési típusukkal, törzsükkel).



Minden olyan elem, amelyet egy class nyelvi elem hatókörében adunk meg.



Osztály neve, és az osztály adattagjainak megadása nevükkel és típusukkal, és a metódusok szignatúráinak (név, paraméterlista, visszatérési típus) felsorolása.



Osztály neve, és az osztályból példányosítandó objektumok.

Helytelen

10. kérdés

0 / 1 pont

Mi alapján példányosodik egy objektum, ha osztályában nem definiálunk konstruktort?



Nem példányosodik: a futtatás során a `NoConstructorException` kivétel keletkezik.



Ilyenkor az objektumnak egy úgynevezett üres sémája jön csak létre, amely object típusú, amelyhez megfelelő kasztolás után lehet hozzáadni az inicializált adattagokat.



Ilyenkor az alapértelmezett üres konstruktor végzi a példányosítást, amely sorra meghívja az adattagok üres konstruktorait.



Ilyenkor az alapértelmezett üres konstruktor végzi a példányosítást, feltéve, hogy az adattagokat leíró osztályokban sem definiáltunk konstruktorokat.

Kvízeredmény: **7** az összesen elérhető 10 pontból