

Programozás alapja 2.	1. beszámoló dolgozat. 2017.02.23.	Kurz/ Terem : Gy1/	Elért pontszám:
Név:	Neptun:		

Az első feladatnál minden bejelölt válasz 1 pont, ha helyes, -1 pont, ha hibás! Az esetleges negatív eredmény is összeadódik a többi feladatra kapott pontokkal. A teljes dolgozat eredménye azonban nem lehet negatív.

1. Feladat

(4p)

Jelölje, hogy mely kijelentés(ek) igaz(ak) a C++ nyelvre!

- | | |
|--|--|
| <input type="checkbox"/> Minden programot <code>using namespace std;</code> direktívával kell kezdeni. | <input type="checkbox"/> Függvényprototípus használata kötelező, ha a függvény a használat előtt nincs definiálva. |
| <input type="checkbox"/> Referencia típusú változó nem adható át paraméterként. | <input type="checkbox"/> A throw utasítás végtelen ciklusból való kilépésre való. |
| <input checked="" type="checkbox"/> A new képes kivételt generálni. | <input type="checkbox"/> Kivételt csak abban a fájlban lehet elfogni, ahol a throw utasítás van. |
| <input type="checkbox"/> Default paramétere csak tagfüggvénynek lehet. | <input type="checkbox"/> Destruktornak csak konstans paramétere lehet. |
| <input checked="" type="checkbox"/> A névterek egymásba ágyazhatók. | <input type="checkbox"/> Konstans tagfüggvény csak konstans paramétert kaphat. |
| | <input type="checkbox"/> Minden programot <code>#include <iostream></code> direktívával kell kezdeni. |

Jelölje, hogy mely kijelentés(ek) igaz(ak) az alábbi C++ kódrészletre!

```
{ int i; while (std::cin >> i); }
```

- Addig olvas be egész számokat, amíg az inputon egész formátumnak megfelelő adatok érkeznek
- A kódrészlet hibás, mert az `i` változó nem kap értéket.
- Végtelen ciklus.

2. **Tervezz** egy olyan osztályt (Ido), ami idő (óra, perc) tárolására alkalmas! Az osztálynak

(6p)

- legyen paraméter nélkül hívható konstruktora, ami 12 óra 0 percet állít be;
- legyen olyan beállító függvénye (set), amivel az óra és a perc is beállítható; amennyiben csak az órát adják meg paraméterként, úgy adott óra 0 percet állítson be a függvény;
- legyenek olyan lekérdező függvényei, amelyekkel az óra és a perc külön-külön is lekérdezhető;
- legyen olyan operátora (!=), amivel el lehet dönteni, hogy két időpont eltér-e;

Valósítsa meg az osztályt C++ nyelven úgy, hogy az osztálynak legyen legalább 1 nem inline tagfüggvénye! Ne lehessen az adattagokhoz kívülről közvetlenül hozzáférni. A beállító függvény (set) dobjon **const char *** hibát, ha hibás paraméterrel hívják (pl. negatív perc)!

Működjön helyesen az alábbi kódrészlet:

```
Ido t0, t10, t13;
t10.set(10); // 10 óra 00 percet állít be
t13.set(13,21); // 13 óra 21 percet állít be
if (t13 != t10) std::cout << "elter"; // elter-t ír ki
```

```
class Ido {
    int ora;
    int perc;
public:
    Ido() :ora(12), perc(0) {}
    void set(unsigned int o, unsigned int p = 0);
    int getOra() const { return ora; }
    int getPerc() const { return perc; }
    bool operator!=(const Ido& rhs) const;
};

void Ido::set(unsigned int o, unsigned int p) {
    if (o > 23 || p > 59) throw "Hiba";
    ora = o;
    perc = p;
}

bool Ido::operator!=(const Ido& rhs) const {
    return ora != rhs.ora || perc != rhs.perc;
}
```

