

III. tétel (30 pont)

Az 1-es feladat esetén, a helyes válasznak megfelelő betűt, írástok a vizsgalagra.

1. Ha rendelkezésre áll a 0, 1 és a 2 -es számjegy, akkor elő lehet állítani olyan növekvő sorrendbe lévő számokat, melyekre igaz az, hogy bármelyik szám számjegyeinek összege 2. Így, az első 6 szám a sorozatból: 2, 11, 20, 101, 110, 200. Ugyanezt az algoritmust használva, előállítják azokat a 0, 1, 2 és 3 -as számjegyekből álló számokat, amelyekre igaz az, hogy a számjegyeik összege 4 -el egyenlő. Melyik lesz a 7-ik szám az így előállított sorozatban? **(4p.)**
- a. 130 b. 301 c. 220 d. 103

Az alábbi feladatok esetén a kérésekre, kérdésekre a választ írástok a vizsgalagra!

2. Tekintsük az `f` alprogram mellékelt definícióját.
- Milyen értéket fog kiírni az `f(14)`; hivatkozás során?

```
void f(int x)
{
    if (x<=10)
        cout<<0<<" "; | printf("%d ",0);
    else
    { f(x-2);
      cout<<x<<" "; | printf("%d ",x);
    }
}
```
3. Az `ordonare` alprogram az `x` paraméteren keresztül egy egydimenziós tömböt kap, mely legtöbb 100 valós számot tartalmazhat, és az `n` paraméterben egy egész értéket kap, mely az `x` tömb elemeinek tulajdonképpen száma. Az alprogram növekvő sorrendbe rendezi a kapott sorozat elemeit, és ugyancsak az `x` paraméteren keresztül, rendelkezésre bocsátja a rendezett tömböt.
- a) Az alprogramnak, csak a fejlécét írástok le, a szükséges típus deklarációkkal együtt. **(4p.)**
- b) Írástok egy C/C++ programot mely beolvas a billentyűzetről két természetes számot, azaz az `n` és az `m` ($1 \leq n \leq 100$ és $m \leq n$) értékét, majd beolvas `n` darab, egymástól különböző, valós számot. Használva az `ordonare` alprogram helyes meghívását, a program kiírja a képernyő első sorába az `m` darab legnagyobb elemet (értékük szerint növekvő sorrendben), és a képernyő második sorába, az `m` darab legkisebb elemet (értékük szerint csökkenő sorrendbe) a beolvasott sorozatból. Ugyanabba a sorba írt értékek, szóközzel lesznek elválasztva. **(10p.)**
- Példa:** ha `n=9`, `m=3`, és a sorozat (14.2, 60, -7.5, -22, 33.8, 80, 4, 10, 3), akkor a képernyőre kiírja a következő két sort:
- 33.8 60 80
- 3 -7.5 -22
4. Írástok egy C/C++ programot, amely létrehozza a `SIR.TXT` szöveges állományt, és beleírja az összes olyan karakterláncot, amelyeket az angol abc két különböző negybetűjéből alkothatunk úgy, hogy egyetlen karakterláncban se szerepeljen egymás mellett két magánhangzó. Mindegyik karakterláncot az állomány külön sorába kell íratni. **(6p.)**