

Szkriptnyelvek - JavaScript ismertető

- A programot JavaScript nyelven kell megírni.
- A benyújtandó fájl neve: `feladat.js`
 - Egy JavaScript nyelven írt, szöveges fájl (nem zip, rar, stb.)
 - Ez csak a feladatban kért dolgokat tartalmazza! Amennyiben saját inputtal teszteled a kódot lokálisan, úgy feltöltés előtt a tesztelő kódrészletet kommenteld ki!
- A megoldást Bíró2 webes felületén (<https://biro2.inf.u-szeged.hu>) keresztül kell benyújtani és a megoldást a Bíró fogja kiértékelni.
 - A Feladat beadása felületen a *Feltöltés* gomb megnyomása után ki kell várni, amíg lefut a kiértékelés. **Kiértékelés közben nem szabad az oldalt frissíteni vagy a Feltöltés gombot újból megnyomni** különben feltöltési lehetőség veszik el!
- Feltöltés után a Bíró a programot **Node** interpreterrel fogja futtatni, és különböző tesztesetekre futtatja.
- A program működése akkor helyes, ha a tesztesetek futása nem tart tovább 2 másodpercnél és hiba nélkül fejeződik be, valamint a program működése a feladatkiírásnak megfelelő.
- Ha 3 teszteset futási ideje túllépi a fenti időkorlátot, a tesztelés befejeződik, a pontszám az addig szerzett pontszám lesz.
- A riport.txt megtekinthető az alábbi módon:
 1. Az Eredmények megtekintése felületen a vizsgálandó próba új lapon való megnyitása
 2. A kapott url formátuma:
https://biro2.inf.u-szeged.hu/Hallg/IB370G/FELADAT_SZAMA/hXXXXXX/4/riport.txt
 3. Az url-ből visszatörölve a 4-esig (riport.txt törlése) megkaphatók a 4-es próbálkozás adatai
- A programot 25 alkalommal lehet benyújtani, a megadott határidőig.
- A munkád során figyelj arra, hogy pontosan kövesd a feladatban leírtakat, az elnevezéseket!
- A fájl elejére kommentbe írd be a neved, Neptun és h-s azonosítód az alábbi formában:

```
// Nev: vezeteknev keresztnév  
// Neptun: NEP4LF  
// h: h123456
```

Szkriptnyelvek versenyfeladat

Fa összeg

Írj JavaScriptben egy `faosszeg` nevű függvényt, amely egy [fában](#) a csúcsok értékeinek összegét adja vissza. \

Paraméterben a fa gyökércsúcsát (egy objektumként) kapjuk meg. \

Egy csúcsnak van

- egy `ertekek` adattagja (szám) (ezeket kell majd összeadni), illetve van
- egy `gyerekek` adattagja, ami a gyerek csúcs objektumokat tartalmazó tömb.

Példa:

```
// egy csúcs
{
  ertek: 10,
  gyerekek: []
}

// egy csúcs egy gyerekkel
{
  ertek: 10,
  gyerekek: [{ertek: 5, gyerekek: []}]
}
```

Számíthatunk rá, hogy a kapott paraméterben minden definiálva van, és minden a megadott típussal rendelkezik, ezeket nem kell leellenőrizni.

A kiértékelő a helyességen kívül futásidőt és a kód méretét is beszámítja a végső pontszámba. Tehát minél gyorsabban fut le a kód, illetve minél kisebb, annál több pontot ér. (A kódméretbe a whitespace karakterek nem számítanak, tehát szóköz, új sor, tab tetszőleges számban lehet benne.)

Tipp: [rekurzív függvény](#).

Példa:

```
const pelda1 = {
  ertek: 10,
  gyerekek: [{ ertek: 5, gyerekek: [] }],
};
console.log(faOsszeg(pelda1)); // 15

const pelda2 = {
  ertek: 5,
  gyerekek: [
    {
      ertek: 3,
      gyerekek: [],
    },
    {
      ertek: 1,
      gyerekek: [
        {
          ertek: 2,
          gyerekek: [
            {
              ertek: 9,
              gyerekek: [],
            },
            {
              ertek: 8,
              gyerekek: [],
            },
            {
              ertek: 4,
              gyerekek: [],
            },
          ],
        },
      ],
    },
  ],
}
```

```
        ],  
      },  
    ],  
  },  
],  
};
```

```
console.log(faosszeg(pelda2)); // 32, mert 5 + 3 + 1 + 2 + 9 + 8 + 4 = 32
```