

1. óra

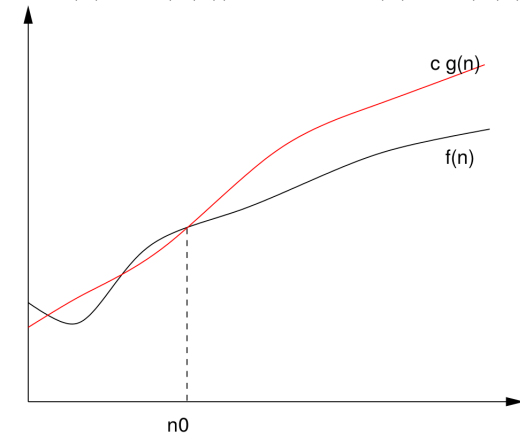
3.2 feladat

Segítség: számtani közép, első n szám összege.

2. óra

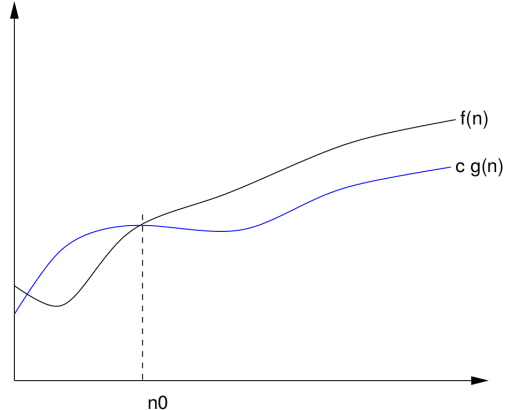
$$\mathcal{O}(g(n)) = \{f(n) : (\exists c, n_0 \geq 0)(\forall n \geq n_0)(0 \leq f(n) \leq cg(n))\}$$

Az $f(n) = \mathcal{O}(g(n))$ jelentése: $f(n) \in \mathcal{O}(g(n))$



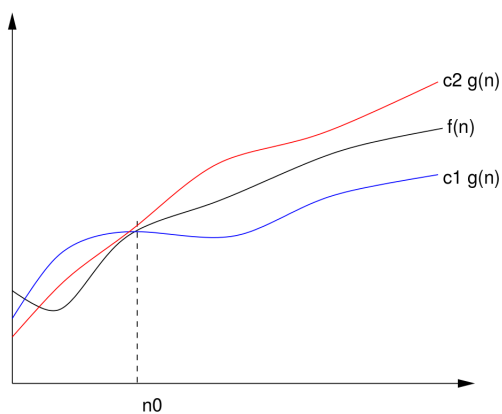
$$\Omega(g(n)) = \{f(n) : (\exists c, n_0 > 0)(\forall n \geq n_0)(cg(n) \leq f(n))\}$$

Az $f(n) = \Omega(g(n))$ jelentése: $f(n) \in \Omega(g(n))$



$$\Theta(g(n)) = \{f(n) : (\exists d_1, d_2, n_0 \geq 0)(\forall n \geq n_0)(d_1 g(n) \leq f(n) \leq d_2 g(n))\}$$

Az $f(n) = \Theta(g(n))$ jelentése: $f(n) \in \Theta(g(n))$



A képek tőle lettek lopva: Németh Tamás

1.2. feladat

3.5. feladat

Segítség: mértani közép

3. óra

Partíció probléma

4.1. feladat

4.2. feladat

Partíció probléma rekurziómemorizálással

4. óra

Dinamikus programozás

Partíció probléma dinamikus táblázatkitöltéssel

7.1. feladat

7.2. feladat

5. óra

4.3., 7.3. feladat

7.17. feladat

4.7. feladat

6. óra

5.4. feladat

6.1. feladat

Megoldás

Megadunk egy rekurzív algoritmust, amit f fa magasságát határozza meg. Az algoritmus azon rekurzív összefüggés alapján számítja ki a fa magasságát, hogy minden csúcs magassága a gyerekei magasságának a maximuma $+1$. (A levelek magassága nyilván 1.)

```
magassag(f)
  max = 0
  g = f.elfoiu
  while (g != NIL)
    m = magassag(g)
    if (m > max)
      max = m
    g = g.testver
  return max + 1
```

6.2. feladat

Megoldás

Megadunk egy rekurzív algoritmust, amit f fa magasságát határozza meg az előző feladatban megadott összefüggés alapján.

```
magassag(f)
  max = 0
  for i = 1 to elemszam(f.fiuk)
    m = magassag(f.fiuk[i])
    if (m > max)
      max = m
  return max + 1
```

6.8. feladat

Megoldás

Megadunk egy rekurzív algoritmust, amit f fa levelienek a számát határozza meg. Az algoritmus azon rekurzív összefüggés alapján számítja ki a fa levelienek a számát, hogy minden csúcs esetén az adott csúcs „alatt található” levelek száma az egyes gyerekei „alatt található” levelek számának az összege. (A leveleken a „levelek száma” nyilván 1.)

```
levelek(f)
```

```

    if (f.elfoiu = NIL)
        return 1
    sum = 0
    g = f.elfoiuk
    while (g != NIL)
        sum = sum + levelek(g)
        g = g.testver
    return sum

```

6.5. feladat

Megoldás

```

KSzint(f, k)
    if (k = 1)
        return 1
    if (f.elfoiu = NIL)
        return 0
    sum = 0
    g = f.elfoiu
    while (g != NIL)
        sum = sum + kszint(g, k-1)
        g = g.testver
    return sum

```

7. óra

Nagy zh

8. óra

Mohó algoritmusok
8.2, 8.3, 8.5

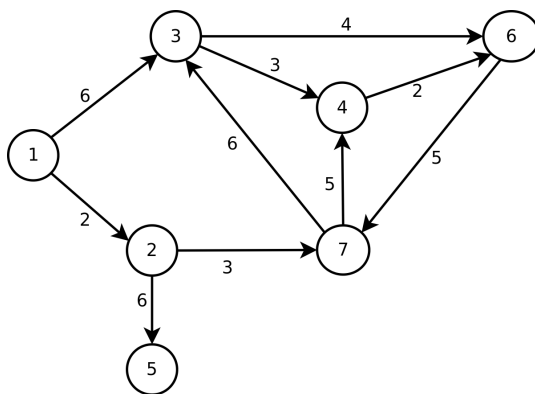
9. óra

Gráfok
10.4, 10.7

10. óra

Erősen összefüggő komponensek, legrövidebb út
11.6, 12.1

Feladat: Az alábbi gráfra az 1 pontból kiindulva hajtsuk végre a Dijkstra algoritmust.



Megoldás

	1	2	3	4	5	6	7
apa	0	1	1	3	2	3	2
d	0	2	6	9	8	10	5

1. táblázat. Dijkstra megoldás

11. óra

14.10
Feszítőfák
13.1, 13.2, 13.4

12. óra

Nagy zh