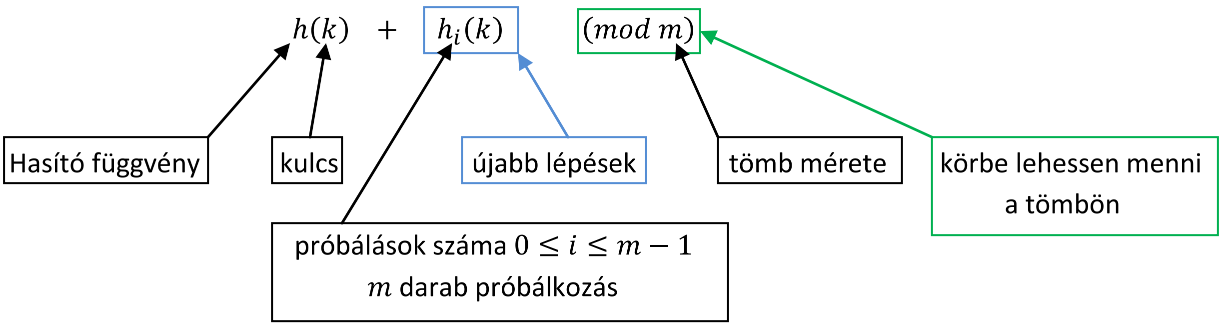
# Algoritmusok és adatszerkezetek 2.

*Fekete István előadása alapján*
*Készítette: Nagy Krisztián*

## 3. előadás

## <u>**20. Hasítás**</u> (folytatás)

### <u>**2.Kulcsütközés feloldása nyitott (nyílt) címzéssel**</u> (folytatás)




$\{h_0(k) \equiv 0, h_1(k), \dots h_{m-1}(k)\} = \{0,1,2,\dots,m-1\} \leftarrow$ lépésszám halmaz

Apróbb ismétlés, ami fontos : A tömb elemeinek / rekordoknak / mezőknek <u>3 státusza van</u>:

a) szabad (üres mező)
b) foglalt
c) <u>törölt</u> → kereséshez elengedhetetlen (általában egy extremális jelet tartalmaz)

**<u>A kulcsütközés feloldásának módszerei</u>:**

1. <u>Lineáris próbálás</u>
   $h_i(k) = -i$
   Tradicionálisan a balra fele lépegetést használjuk, amíg üres helyet nem találunk
   Hátránya: Elsődleges csomósodást eredményez (kevésbé szortan helyezkednek el az elemek)
2. <u>Négyzetes próbálás:</u>
   Matematikai tétel miatt a tömb mérete $(m)$ $m = 4k + 3$ alakú prím legyen.
   A próbasorozat: $0^2, \pm(1^2), \pm(2^2), \pm(3^2), \dots, \pm\left[\left(\frac{n-1}{2}\right)^2\right]$ (, ahol + és – a lépés iránya.
   (Tétel: Minden tömbelemhez eljutunk ezzel a módszerrel)
   Hátrány: Másodlagos csomósodást eredményez. (jobban szór mint az előbbi, de még nem az igazi)
3. <u>Kettős hasítás</u>
   $h_i(k) = -i \cdot h'(k) \quad (mod\ m)$

   relatív prím $m$-hez $\qquad h'(k) = k(mod\ m-1) + 1$
   Példa: $k_1 = 25 \quad k_2 = 36 \qquad h(25) = 25\ mod\ 11 = 3 \qquad h(36) = 36\ mod\ 11 = 3$
   Láthatjuk, hogy kulcsütközés történik. Kiszámoljuk a másodlagos hashelő függvénnyel:
   $h'(25) = 25(mod\ 11 - 1) + 1 = 25(mod\ 10) + 1 = 6$
   $h'(36) = 36(mod\ 11 - 1) + 1 = 36(mod\ 10) + 1 = 7$ Itt már nincs csomósodás!

# 3.Hasító függvények

- Tapasztalat „folklórja”, kreativitás
- Elmélet ⇒ bizonyítható tulajdonságok

A. Osztómódszer

$h(k) = k\ (mod\ m)$

Javaslat (tapasztalat) : Jó, ha $m$ prím és nem esik 2 hatvány közelébe

Például: Van $n = 2000$ rekordunk → $m = 701$ esetén körülbelül 3 hosszú listák alakulnak ki a hasítótáblához csatolva

B. Szorzómódszer

$$h(k) = \lfloor \underbrace{m}_{\text{tábla méret}} \cdot \underbrace{\boxed{(\underbrace{k}_{\text{kulcs}} \cdot \underbrace{A}_{0<A<1}\ mod\ 1)}}_{\text{törtrész}} \rfloor$$

Például $A = \frac{\sqrt{5}-1}{2} \approx 0{,}618$ az aranymetszésből jól ismerhető képlet megfelelő választás A-ra. Ekkor egyenletes lesz az eloszlás.

Ha $m = 2^l$, akkor könnyen számolható a képlet. Lásd Algoritmusok könyv.

*[Kitekintés: Algoritmusok 3,4 MSc-s tárgyból bővebben tanulhatjuk. (Nem vizsga anyag)*

*Univerzális hasítás:*

$h_{a,b}(k) = \big((a \cdot k + b)\ mod\ p\big)\ mod\ m$     *p- prím szám*

$1 \le a \le p-1$        $0 \le b \le p-1$

*Két tulajdonság (tétel):*

1. *Véletlen (egyenes eloszlású) kulcssorozatot várhatóan egyenletesen hasít*
2. *Bármilyen (rossz) kulcssorozatot várhatóan egyenletesen hasít egy véletlen a és b választással képzett h függvény*

*]*

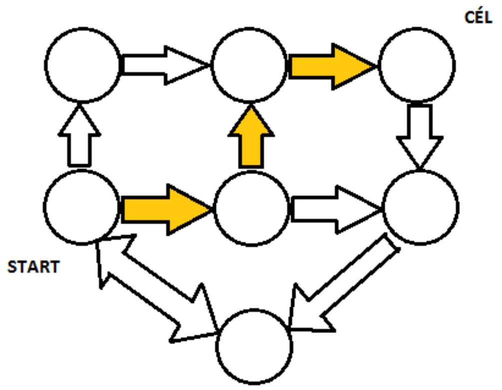
# VI. GRÁFALGORITMUSOK

## Bevezetés:

Gráf matematikai fogalom ⇒ adatszerkezet
ADS Ábrázolás

Kognitív sémák!

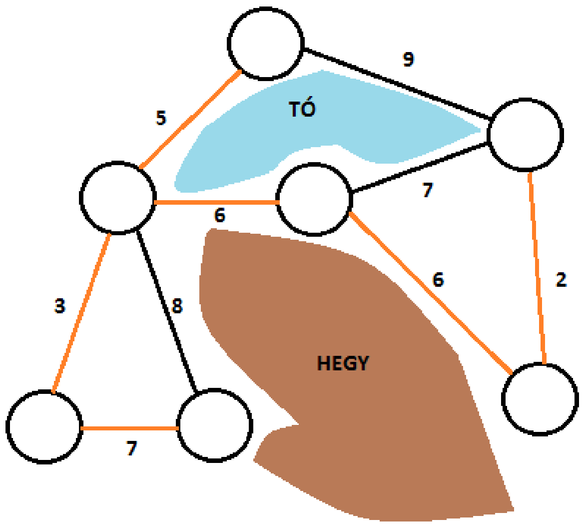
Két példa:
1. Autós útkeresés Budapesten (irányított, súly nélküli gráf)
Van egy START csúcs : akkor irányított gráf általában




Dijkstra-algoritmus
(Mesterséges intelligeciából majd az $A, A^{+}$ kereső algoritmus)

2. Minimális költségű feszítőfa (irányítás nélküli, súlyozott gráf)

Például: Városokba az elektromos áram bevezetése. Hogy hatékonyabb költségben (kisebb méretű távok esetén kevesebb kábel kell)




Kruskal – algoritmus
Prim – algoritmus

# 21. Gráfok ábrázolása

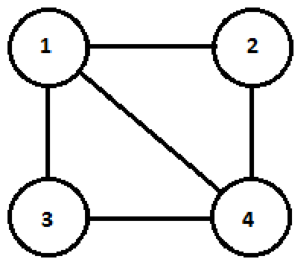
## Alapfogalmak:

(Felsorolás szerűen. Diszkrét matematika 2. kurzusból már tanultuk őket)

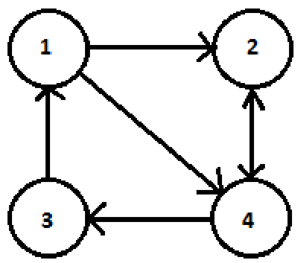
- irányított gráf
- irányítás nélküli gráf
- szomszéd/rákövetkező
- út fogalma
- fokszám, kifok és befok
- összefüggőség
- teljes gráf
- páros gráf
- erdő
- Élsúly: $c: E \to \mathbb{R}$
- $n = |V|$
- $e = |E|$

## 1.Csúcsmátrix

Mivel nem engedjük, hogy legyen hurokél, ezért a főátló az 0 lesz.

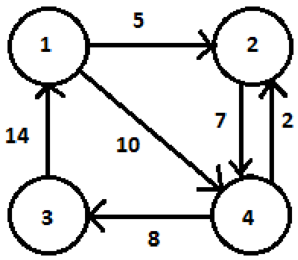
| | | |
|---|---|---|
| $(A)$ |  | $G = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$ |

Mivel irányítatlan, így a mátrix a főátlóra szimmetrikus.

| | | |
|---|---|---|
| $(B)$ |  | $G = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$ |

Élsúlyok esetén:

- Ahol van él ⇒ a mátrixba az élsúlyát írjuk
- Ahol nincs él ⇒ a mátrixba $+\infty$-t írunk
- A mátrix főátlójába 0-t írunk, mivel alapjáraton nem lehet hurokél

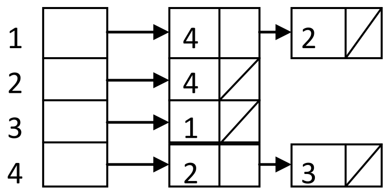
| | | |
|---|---|---|
| $(C)$ |  | $G = \begin{pmatrix} 0 & 5 & \infty & 10 \\ \infty & 0 & \infty & 7 \\ 14 & \infty & 0 & \infty \\ \infty & 2 & 8 & 0 \end{pmatrix}$ |

<u>Alkalmazás:</u> Sűrű gráfok esetén: $e = n^2$

## <u>2.Éllistás ábrázolás</u>

### $(\boldsymbol{B})$ eset:

Ez az általános ábrázolás!




$Adj[1..n]$ - adjacencia (szomszédosság)

Helyfoglalás: $Mem(n, e) = \theta(n + e)$
Alkalmazás: Normál / ritka gráfok esetén Normál: $e \sim n$ Például: 5000 él, 1000 csúcs

### $(\boldsymbol{A})$ eset:

Az előzőhöz képest több elemet tartalmaz $(u, v)$ *és* $(v, u)$is szerepel az irányítatlanság miatt.
Helyfoglalás: $Mem(n, 2e) = \theta(n + e)$

### $(\boldsymbol{C})$ eset:

Az egyes listaelemek bővülnek egy mezővel, amiben az élsúlyokat tároljuk.
Részlet: