10. Gráf absztrakt adattípus, gráfok ábrázolása

10.1. Definíciók

1. **Irányítatlan gráf**:$G = (V,E)$
E rendezetlen $\{a,b\}, a,b \in V$ párok halmaza.
2. **Irányított gráf**:$G = (V,E)$
E rendezett (a,b) párok halmaza; $E \subseteq V \times V$.
3. **Multigráf**:
$G = (V,E,\mathit{Ind},\mathit{Érk})$, $\mathit{Ind},\mathit{Érk} : E \to V$
$\mathit{Ind}(e)$ az e él induló, $\mathit{Érk}(e)$ az érkező pontja.
Címkézett (súlyozott) gráf:$G = (V,E,C)$
$C : E \to \mathit{Címke}$

$Ki(G,p) = \{q \in V : (p,q) \in E\}$
$Be(G,p) = \{q \in V : (q,p) \in E\}$
$KiFok(G,p) = |Ki(G,p)|$
$BeFok(G,p) = |Be(G,p)|$

10.2. Gráf absztrakt adattípus

Milyen műveleteket akarunk gráfokon végezni?

Értékhalmaz: $Graf = \{G = (V,E) : V \subseteq PontTip, E \subseteq V \times V\}$
Műveletek:

$$G : Graf,\ P, P1, P2 : PontTip,\ I : PIterator,$$

$\{Igaz\}$	$Letesit(G)$	$\{G = (\emptyset,\emptyset)\}$
$\{G = G\}$	$Megszuntet(G)$	$\{Igaz\}$
$\{G = G\}$	$Uresit(G)$	$\{G = (\emptyset,\emptyset)\}$
$\{G = (V,E)\}$	$PBovit(G,P)$	$\{V = Pre(V) \cup \{P\} \wedge E = Pre(E)\}$
$\{G = (V,E) \wedge P \in V\}$	$PTorol(G,P)$	$\{V = Pre(V) - \{P\} \wedge$
		$E = Pre(E) - \{(P,Q) : Q \in KI(G,P)\} -$
		$\{(Q,P) : Q \in Be(G,P)\}\}$
$\{G = (V,E), P1, P2 \in V\}$	$ElBovit(G,P1,P2)$	$\{E = Pre(E) \cup \{(P1,P2)\} \wedge V = Pre(V)\}$
$\{G = (V,E), P1, P2 \in V\}$	$ElTorol(G,P1,P2)$	$\{E = Pre(E) - \{(P1,P2)\} \wedge V = Pre(V)\}$
$\{G = (V,E), P1, P2 \in V\}$	$Vanel(G,P1,P2)$	$\{= (P1,P2) \in E \wedge E = Pre(E) \wedge V = Pre(V)\}$
$\{G = G\}$	$PIterKezd(G,I)$	$\{\}$
$\{G = G\}$	$PIterAd(I,P)$	$\{\}$
$\{G = G\}$	$PIterVege(I)$	$\{\}$
$\{G = G\}$	$KiIterKezd(G,P,I)$	$\{\}$
$\{G = G\}$	$PKiIterAd(I,Q)$	$\{\}$
$\{G = G\}$	$KiIterVege(I)$	$\{\}$
$\{G = G\}$	$BeIterKezd(G,P,I)$	$\{\}$
$\{G = G\}$	$BeIterAd(I,P)$	$\{\}$
$\{G = G\}$	$BeIterVege(I)$	$\{\}$

$$\{G=G\} \quad ElIterKezd(G,I) \quad \{\}$$
$$\{G=G\} \quad ElIterAd(I,P,Q) \quad \{\}$$
$$\{G=G\} \quad ElIterVege(I) \quad \{\}$$

`Forall p in V Do`
`  M(p);`
	≡	`PIterKezd(G,I);`
		`While Not PIterVege(I) Do Begin`
		`  PIterAd(I,p);`
		`  M(p); {művelet a p ponttal}`
		`End;`

`Forall q in Ki(G,p) Do`
`  M(p,q);`
	≡	`KiIterKezd(G,p,I);`
		`While Not KiIterVege(I) Do Begin`
		`  KiIterAd(I,q);`
		`  M(p,q); {művelet a p->q éllel}`
		`End;`

`Forall p in Be(G,q) Do`
`  M(p,q);`
	≡	`BeIterKezd(G,q,I);`
		`While Not BeIterVege(I) Do Begin`
		`  BeIterAd(I,p);`
		`  M(p,q); {művelet a p->q éllel}`
		`End;`

`Forall (p,q) in E Do`
`  M(p,q);`
	≡	`ElIterKezd(G,I);`
		`While Not ElIterVege(I) Do Begin`
		`  ElIterAd(I,p,q);`
		`  M(p,q); {művelet a p->q éllel}`
		`End;`

Az él-iteráció nyilvánvalóan megvalósítható a pont-iteráció és ki-iteráció műveletekkel, de fordítva nem.

```
Forall (p,q) in E Do          ≡      PIterKezd(G,Ip);
  M(p,q);                            While Not PIterVege(Ip) Do Begin
                                       PIterAd(Ip, P);
                                       KiIterKezd(G, P, Ie);
                                       While Not KiIterVege(Ie) Do Begin
                                         KiIterAd(Ie,q);
                                         M(p,q); {művelet a p->q éllel}
                                       End{while Ie}
                                     End{while Ip};
```

Címkézett (súlyozott) gráf absztrakt adattípus

Értékhalmaz:

$Graf = \{G = (V,E,C) : V \subseteq PontTip, E \subseteq V \times V, C : E \to CimkeTip\}$

Műveletek:

$$G : Graf,\ P, P1, P2 : PontTip,\ S : CimkeTip,\ I : PIterator,$$

$$
\begin{array}{rcl}
\{Igaz\} & Letesit(G) & \{G=(\emptyset,\emptyset)\} \\
\{G=G\} & Megszuntet(G) & \{Igaz\} \\
\{G=G\} & Uresit(G) & \{G=(\emptyset,\emptyset)\} \\
\{G=(V,E)\} & PBovit(G,P) & \{V=Pre(V)\cup\{P\}\wedge E=Pre(E)\} \\
\{G=(V,E)\wedge P\in V\} & PTorol(G,P) & \{V=Pre(V)-\{P\}\wedge \\
 & & E=Pre(E)-\{(P,Q):Q\in KI(G,P)\}- \\
 & & \{(Q,P):Q\in Be(G,P)\}\} \\
\{G=(V,E),P1,P2\in V\} & ElBovit(G,P1,P2,S) & \{E=Pre(E)\cup\{(P1,P2)\}\wedge C(P1,P2)=S\} \\
\{G=(V,E),P1,P2\in V\} & ElTorol(G,P1,P2) & \{E=Pre(E)-\{(P1,P2)\}\wedge V=Pre(V)\} \\
\{G=(V,E),P1,P2\in V\} & Vanel(G,P1,P2) & \{=(P1,P2)\in E\wedge G=Pre(G)\} \\
\{G=(V,E),(P1,P2)\in E\} & ElCimke(G,P1,P2,S) & \{S=Pre(C)(P1,P2)\in E\wedge E=Pre(E)\} \\
\{G=(V,E),(P1,P2)\in E\} & ElCimkez(G,P1,P2,S) & \{S=C(P1,P2)\wedge E=Pre(E)\} \\
\{G=G\} & PIterKezd(G,I) & \{\} \\
\{G=G\} & PIterAd(I,P) & \{\} \\
\{G=G\} & PIterVege(I) & \{\} \\
\{G=G\} & KiIterKezd(G,P,I) & \{\} \\
\{G=G\} & PKiIterAd(I,Q,S) & \{\} \\
\{G=G\} & KiIterVege(I) & \{\} \\
\{G=G\} & BeIterKezd(G,P,I) & \{\} \\
\{G=G\} & BeIterAd(I,P,S) & \{\} \\
\{G=G\} & BeIterVege(I) & \{\}
\end{array}
$$

$$
\begin{array}{ccc}
\{G=G\} & ElIterKezd(G,I) & \{\} \\
\{G=G\} & ElIterAd(I,P,Q,S) & \{\} \\
\{G=G\} & ElIterVege(I) & \{\}
\end{array}
$$

10.3. Gráfok ábrázolásai

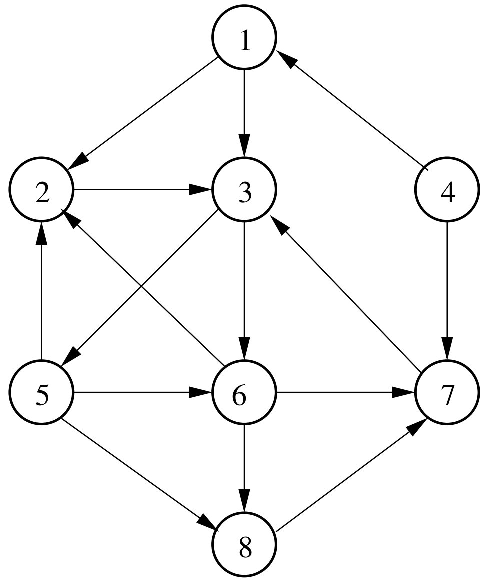
Szempontok az adatszerkezet megválasztásához.

1. Az adott probléma megoldásához ténylegesen mely műveletek szükségesek.
2. Melyek a releváns műveletek, amelyek alapvetően befolyásolják az algoritmus futási idejét.
3. A tárigény az adott probléma esetén.

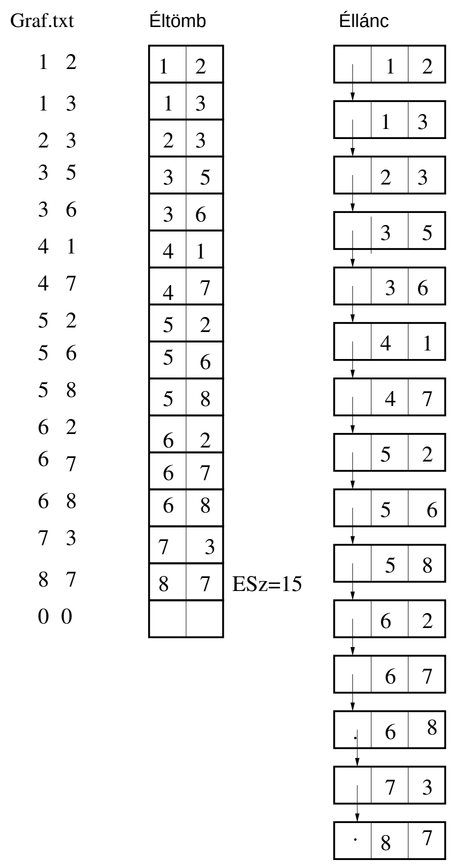
1. **Élhalmaztömb és lánc**

a.) Statikus (tömbös)

```
Const MaxE=???; MaxN=???
Type
   PontTip=1..MaxN;
   Graf=Array[1..MaxE] of Record
                              Ind,Erk:PontTip
                           End;
```



1. ábra. Példa gráf



2. ábra. Élhalmaztömb és lánc

Tárigény: $\Theta(|E|)$
VanEl időigénye: $\Theta(|E|)$
El-iteráció időigénye: $\Theta(|E|)$
Ki-iteráció időigénye: $\Theta(|E|)$

b.) Dinamikus

```
Type
   PontTip=???;
   Lanc=^Cella;
   Cella=Record
              Ind,Erk:PontTip;
              Csat:Lanc
           End;
   Graf=Lanc
```

Tárigény: $\Theta(|E|)$
VanEl időigénye: $\Theta(|E|)$
El-iteráció időigénye: $\Theta(|E|)$
Ki-iteráció időigénye: $\Theta(|E|)$

2. **Kapcsolatmátrix (szomszédsági mátrix)**

G	1	2	3	4	5	6	7	8	9
1		1	1						
2			1						
3					1	1			
4	1						1		
5		1				1		1	
6		1					1	1	
7			1						
8							1		
9									

3. ábra. Kapcsolatmátrix

```
Const
   MaxN=???
   NemEl=???
Type
   PontTip=1..MaxN;
   Graf=Array[1..MaxN, 1..MaxN] of Boolean;
   GrafC=Array[1..MaxN, 1..MaxN] of Cimke;
```

Tárigény: $\Theta(n^2)$ ($n = |V|$)
VanEl időigénye: $\Theta(1)$
El-iteráció időigénye: $\Theta(n^2)$
Ki-iteráció időigénye: $\Theta(n)$

3. **Éllista**

a.) Statikus

G	1	2	3	4	5	6	7	8	9	KiFok
1	2	3	0							2
2	3	0								1
3	5	6	0							2
4	1	7	0							2
5	2	6	8	0						3
6	2	7	8	0						3
7	3	0								1
8	7	0								1
9	0									0

4. ábra. Statikus éllista

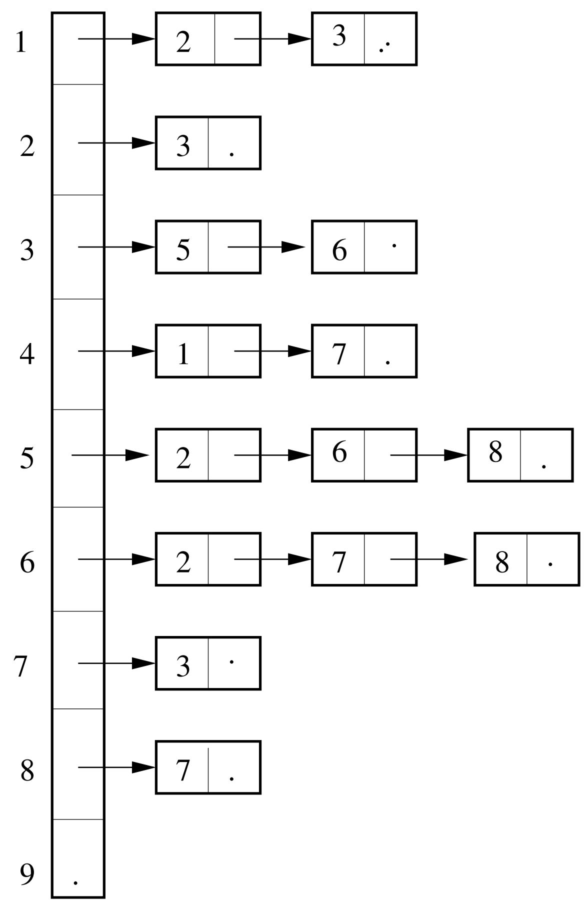
```
Const
   MaxN=???
Type
   PontTip=1..MaxN;
   Graf=Array[1..MaxN, 1..MaxN] of PontTip;
Var
   KiFok:Array[1..MaxN] of 0..MaxN;
   G:Graf;
```

Tárigény: $\Theta(n^2)$
VanEl időigénye: $T_{lr} = O(n)$
El-iteráció időigénye: $\Theta(|E|)$
Ki-iteráció időigénye: $\Theta(|Ki(G,p)|)$

b.) Dinamikus

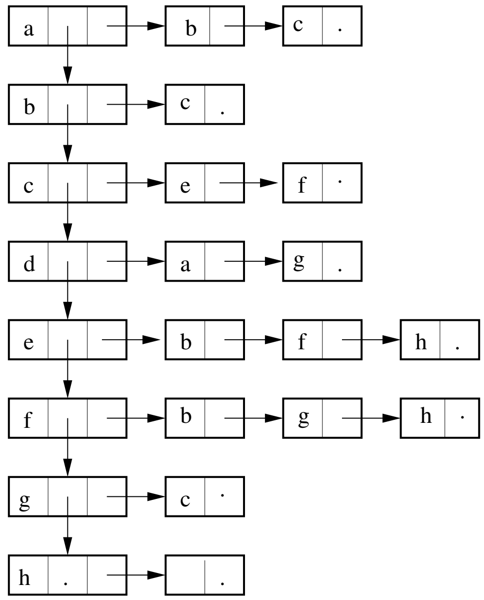
```
Const
   MaxN=???
Type
   PontTip=1..MaxN;
   Graf=Array[1..MaxN] of Lanc;
   Lanc=^Cella;
   Cella=Record
            Erk:PontTip;
            Csat:Lanc
         End;
```

Tárigény: $\Theta(|V|+|E|)$
VanEl időigénye: $T_{lr} = O(n)$
El-iteráció időigénye: $\Theta(|E|)$
Ki-iteráció időigénye: $\Theta(|Ki(G,p)|)$



5. ábra. Dinamikus éllista

4. **Általános éllista**



6. ábra. Általános éllista

```
Type
   PontTip=???;
   FLanc=^FCella;
   VLanc=^VCella;
   FCella=Record
            Ind:PontTip;
            FCsat:FLanc
            VCsat:VLanc
         End;
   VCella=Record
            Erk:PontTip;
            Csat:VLanc
         End;
   Graf=FLanc;
```

Tárigény: $\Theta(|V|+|E|)$
VanEl időigénye: $T_{lr} = O(n)$
El-iteráció időigénye: $\Theta(|E|)$
Ki-iteráció időigénye: $T_{lr} = O(n) + \Theta(|Ki(G,p)|)$

5. **Halmazfüggvény**

$$G = (V,E), \quad E \subseteq V \times V$$

$$r_E : V \to 2^V \quad r_E(p) = Ki(G,p) = \{q : (p,q) \in E\}$$

```
Const
   MaxN=???
Type
   PontTip=1..MaxN;
```

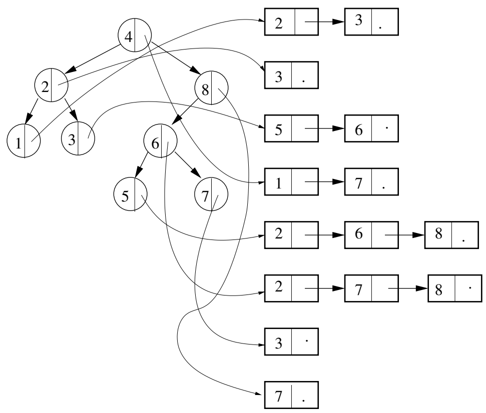
```
Graf=Array[1..MaxN] of Set of 1..MaxN;
```

VanEl(G,p,q) megvalósítása $O(1)$ idejű:
$VanEl(G,p,q) = qInG[p]$
KiIterátor megvalósítása $\Theta(n)$ idejű:

```
For q:=1 To N Do
   If q  In  G[p] Do
       M(p,q);
```

6. **V keresőfában, E (láncban) keresőfában**

Kiegyensúlyozott keresőfa használatával a műveletek futási ideje:



7. ábra. Keresőfás ábrázolás

Tárigény: $\Theta(|V|+|E|)$
VanEl időigénye: $T_{lr} = O(\lg n)$
El-iteráció időigénye: $\Theta(|E|)$
Ki-iteráció időigénye: $O(\lg n) + \Theta(|Ki(G,p)|)$

7. **Dinamikus változók összekapcsolása** (Pl. fák)

```
Type
   PontTip=???;
   Graf=^Pont;
   Lanc=^ElCella;
   Pont=Record
             Ind:PontTip;
             El:Lanc
          End;
   ElCella=Record
             El:Graf;
             Csat:Lanc
          End;
```

8. **Számított gráf**

Az élek halmazát explicite nem tároljuk, mert van olyan számítási eljárás, amely bármely két $p,q \in V$-re kiszámítja $VanEl(p,q)$-t. Vagy, van olyan számítási eljárás, amely minden $p \in V$-re kigenerálja a $Ki(G,p)$ halmaz elemeit.

10.4. Elemi gráfalgoritmusok

10.4.1. Utak

Legyen $G=(V,E)$ irányított (irányítatalan) gráf.
Def. $p,q \in V$-ra egy p-ből q-ba vezető út G-ben:
jele: $\pi : p \rightsquigarrow q$, olyan $\pi = \langle p_0, p_1, \cdots, p_k \rangle$ pontsorozat, ahol
$p_i \neq p_j$, ha $i \neq j$ és $p = p_0$ és $q = p_k$, továbbá
$p = q = p_0$, vagy ($\forall i \in 1..k$) $(p_{i-1}, p_i) \in E$
Def. A $\pi = p \rightsquigarrow q$ út hossza, $|\pi| = |p \rightsquigarrow q| = k$
Def. p-ből q-ba vezető legrövidebb út hossza,
p és q távolsága:

$$\delta(p,q) = \begin{cases} \infty & \texttt{ha nincs } p \rightsquigarrow q \\ Min\{|p \rightsquigarrow q|\} & \texttt{ha van } p \rightsquigarrow q \end{cases} \tag{1}$$

A $\pi = \langle p_0, p_1, \cdots, p_k \rangle$ pontsorozatot a p_0-ból p_k-ba vezető *sétának* nevezzük, ha $\forall i \in 1..k$) $(p_{i-1}, p_i) \in E$

Ha $G=(V,E,C)$ élei a $C: E \to \mathbb{R}$ fg.-el
"súlyozottak", akkor $|p \rightsquigarrow q| = \sum_{i=1}^{k} C(p_{i-1}, p_i)$

Útproblémák

1. Adott $p,q \in V$-re van-e $p \rightsquigarrow q$ út?

2. Adott p-re az Elér$(p) = \{q : p \rightsquigarrow q\}$ halmaz kiszámítása.

3. Adott $p,q \in V$-re $\delta(p,q)$ és egy $p \rightsquigarrow q$ l.r. út kiszámítása.

4. Egy pontból induló legrövidebb utak : adott p-re minden q-ra $\delta(p,q)$ és egy $p \rightsquigarrow q$ l.r. út kiszámítása.

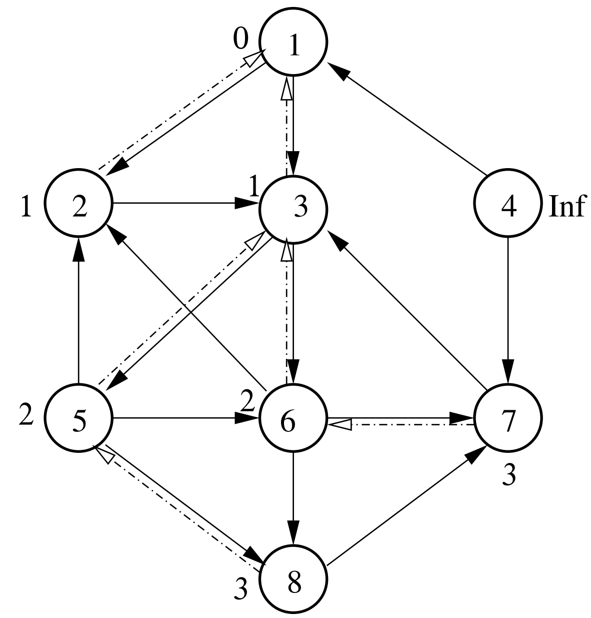
5. Minden p,q pontpárra $\delta(p,q)$ és egy $p \rightsquigarrow q$ l.r. út kiszámítása.

10.5. Szélességi keresés

```
 Procedure SzeltKereres(Const G: Graf.Tipus;
                             S: Graf.PontTip;
                       Var Apa: Fuggveny1;
                         Var D: Fuggveny2);
  Const
    Null=???; {Graf.PontTip, a definiálatlan}
    Inf =???;   {a végtelen reprezentánsa}
  Var
    Q : Sor;
    u,v : Graf.Ponttip;
Begin{SzeltKeres}
  Fuggveny1.Letesit(Apa);
  Sor.Letesit(Q);
  Fuggveny2.Letesit(D);
  For v In V Do Begin
    D[v]:=Inf;
    Apa[v]:=Null;
  End;
  D[S]:=0;
  Sorba(Q,S);

  While Sor.Elemszam(Q)<>0 Do Begin
```

```
    Sor.Bol(Q,u);
    For v In Ki(G,u) Do      {minden u-ból induló élre}
      If D[v]=Inf Then Begin {v érintetlen pont}
        D[v]:=D[u]+1;
        Apa[v]:=u;
        Sorba(Q,v)
      End;
  End{while};
End {SzeltKereres};
```



8. ábra. A példa gráf szélességi bejárása az 1 pontra

1. Lemma
Legyen $G = (V,E)$ irányított vagy irányítatlan gráf és $s,u,v \in V$.
Minden $(u,v) \in E$ élre
$\delta(s,v) \leq \delta(s,u)+1$

2. Lemma
Ha SZELTKERES algoritmust az s pontra alkalmazzuk, akkor a kiszámított D-re teljesül
$(\forall v \in V)\,(D[v] \geq \delta(s,v))$
Biz. Az Q sorba kerülés szerinti indukcióval.
i) Az első pont, ami bekerül: S, de $D[S] := 0$ és $\delta(S,S) = 0$
ii) Tfh. $Sorbol(Q,u)$ után az $u \rightarrow v$ élet vigzsgáljuk és $D[v] = Inf$. Ekkor

$$\begin{aligned} D[v] &= D[u]+1 \\ &\geq \delta(S,u)+1 \\ &\geq \delta(S,v) \end{aligned}$$

és v-t a sorba rakjuk.

3. Lemma
Legyen a SZELTKERES algoritmust végrehajtásának egy pillanatában az $Q =< v_1, v_2, \ldots, v_r >$. Ekkor
$D[v_r] \leq D[v_1]+1$ és $D[v_i] \leq D[v_{i+1}]$ $(i = 1,\ldots,r-1)$
Következmény
Ha egy u pont előbb kerül a sorba, mint a v pont, akkor $D[u] \leq D[v]$.

Biz.
a) $Sorbol(S,u)$ után: $Q =< v_2,\dots,v_r >$
Bizonyítandó: $D[v_r] \le D[v_2]+1$
de $D[v_r] \le D[v_1]+1 \le D[v_2]+1$
b) $Sorba(S,v)$ után: $Q =< v_2,\dots,v_r,v >$
Bizonyítandó: $D[v_r] \le D[v]$ és $D[v] \le D[v_2]+1$
$D[v_r] \le D[v_1]+1 = D[v]$
$D[v_1] \le D[v_2] \Rightarrow D[v] := D[u=v_1]+1 \le D[v_2]+1$

Tétel. $\forall v \in V\ (\delta(S,v) = D[v])$
Biz.
a) $\delta(S,v) = \infty$ azaz nincs $S \rightsquigarrow v$ út.
Ekkor v nem kerülhet be a sorba, tehát marad a kezdetben kapott $D[v] = Inf$

b) Tfh.$\delta(S,v) \neq \infty$
Indir. biz. : Tfh. v a legkisebb olyan $\delta(S,v)$ értékű pont, hogy $\delta(S,v) \neq D[v]$
($s \neq v$) és 2. lemma miatt: $D[v] > \delta(S,v)$ és $\delta(s,v) < \infty$
Legyen u egy olyan pont, amely közvetlenül megelőzi v-t az $s \rightsquigarrow v$ legrövidebb úton: $S \rightsquigarrow u \rightarrow v$
Mivel $\delta(S,u) < \delta(S,v)$, az ind. feltevés szerint: $D[u] = \delta(S,u)$ és $\delta(s,v) = \delta(S,u)+1$

$D[v] > \delta(S,v) = \delta(S,u)+1 = D[u]+1$

Tekintsük azt a helyzetet, amikor a SZELTKERES algoritmus az $u \rightarrow v$ élet vizsgálja.
I) $D[v] = Inf$: tehát $D[v] := D[u]+1 = \delta(S,v)+1$ Ellentmondás!
II)$D[v] < Inf$: tehát v már korábban kapott
$D[v] < \infty$ értéket.
De $D[v] \le D[u]$, mivel v már nincs a sorban, ami ellentmondás!

A SZELTKERES algoritmus futási ideje:
$T_{lr} = O(V+E)$,
feltéve, hogy a Ki-iteráció lineáris idejű.
Ez teljesül, ha éllistás ábrázolást alkalmazunk.

Feszítőfa
A $G = (V,E)$ (irányítatlan) gráfnak az $F = (\overline{V},\overline{E})$ gráf a $p \in V$ gyökerű feszítőfája, ha
1. F részgráfja G-nek ($\overline{V} \subseteq V, \overline{E} \subseteq E,$) és fa.
2. ($\forall q \in V$) ha van $p \rightsquigarrow q$ G-ben, akkor és csak akkor, ha van $p \rightsquigarrow q$ F-ben.

Állítás.
A $G_\pi = (V_\pi, E_\pi)$ gráf, ahol
$V_\pi = \{u : u \in V \wedge Apa[u] \neq Nill\} \cup \{S\}$
$E_\pi = \{(u,v) : Apa[v] = u \neq Nill\}$
G-nek egy S-gyökerű feszítőfája. Továbbá, G_π az S-ből induló legrövidebb utak feszítőfája.

A SZELTKERES algoritmus megvalósítása éllistás gráf ábrázolás esetére.

```
Const
  MaxN=10000;
Type
  PontTip=1..MaxN;
  Lanc=^Cella;
  Cella=Record
          Erk:PontTip;
          Csat:Lanc
       End;
  Graf=Array[1..MaxN] of Lanc;
  Fuggveny1=Array[1..MaxN] of 0..MaxN;
  Fuggveny2=Array[1..MaxN] of 0..MaxN;
```

```
  Sor=Record End;

 Procedure Sorba(S:Sor;x:Ponttip);
   Begin {???} End;
 Procedure Sorbol(S:Sor;Var x:Ponttip);
   Begin {???} End;
 Procedure Letesit(Var S:Sor);
   Begin {???} End;
 Function Elemszam(Var S:Sor):Word;
   Begin {???} End;
Var N:PontTip;
Procedure SzeltKereres(Const G: Graf;
                             S: PontTip;
                       Var Apa: Fuggveny1;
                         Var D: Fuggveny2);
  Const
    Null=0;    {Graf.PontTip, a definiálatlan}
    Inf =MaxN; {a végtelen reprezentánsa}
  Var
    Q : Sor;
    u,v  : Ponttip;
    vp: Lanc;

Begin{SzeltKeres}
  Letesit(Q);
  For v:=1 To N Do Begin
    D[v]:=Inf;
    Apa[v]:=Null;
  End;
  D[S]:=0;
  Sorba(Q,S);

  While Elemszam(Q)<>0 Do Begin
    SorBol(Q,u);
    vp:=G[u];
    While vp<>NIl Do Begin   {minden u-ból induló élre}
      v:=vp^.Erk;            { az u->v élet vizsgáljuk}
      If D[v]=Inf Then Begin {v érintetlen pont}
        D[v]:=D[u]+1;
        Apa[v]:=u;
        Sorba(Q,v)
      End;
      vp:=vp^.csat; {továbblépés a Ki(G,u) halmaz láncában}
    End;
  End{while};
End {SzeltKereres};
```