uגרף מכוון קשיר: גרף מכוון G עבורו לכל  $u,v \in V$  לכל  $u,v \in V$  איים מסלול מuuים מסלול מ־u ל־u מכוון קשיר חזק: גרף מכוון u עבורו לכל אזי  $s\in V\left(G
ight)$  אזי :BFS אזי אלגוריתם

```
function BFS(G, s):
      (d, \pi, \operatorname{color}) \leftarrow \operatorname{dict}(V(G))
      for u\in V\left( G\right) \backslash \{s\} do
            color[u] \leftarrow White
            d[u] \leftarrow \infty
           \pi[u] \leftarrow Null
      end
      color[s] \leftarrow Grey
      d[s] \leftarrow 0
      \pi[s] \leftarrow \text{Null}
      Q \leftarrow queue()
      while Q \neq \varnothing do
            u \leftarrow \mathsf{Q}.\mathsf{head}
            for v \in Neighbor(u) do
                 if color(v) = White then
                        color[v] \leftarrow Grey
                        d[v] \leftarrow d[u] + 1
                        \pi[v] \leftarrow u
                        Q.enqueue(v)
            end
            O.dequeue()
            color[u] \leftarrow Black
      end
     return (d, \pi, \text{color})
```

```
\mathcal{O}\left(|V|+|E|
ight) הינה BFS (G,s) טענה: יהי אזי סיבוכיות אי סיבוכיות אי s\in V\left(G
ight) הינה מענה:
                                                           \{v \in V \mid \mathtt{BFS}\left(G,s
ight).\mathsf{color}\left[v
ight] = \mathtt{Black}\} = [s]_{
ightarrow} אזי איז s \in V משפט: יהי
                                                           \delta(v,u) = \min(\{\operatorname{len}(\sigma) \mid v,u \mid v \in V \mid u,v \in V \mid u,v \in V \} סיול בין יהי
                                                          \delta\left(v,u
ight) \leq \delta\left(v,w
ight) + 1 אזי \left(w,u
ight) \in E טענה: יהי G גרף ויהיו G גרף באשר באשר
                                                      d\left[v
ight] \geq \delta\left(v
ight) מתקיים BFS \left(G,s
ight) אזי בכל שלב בהרצת אזי גרף ויהיו s,v\in V למה: יהי
             d[v_i] \leq d[v_1]+1 וכן d[v_i] \leq d[v_{i+1}] אזי מתקיים שלב בהרצת BFS (G,s) בו בהרצת G
                                                              .BFS (G,s) .d[v] = \delta\left(v,s\right) אזי s,v \in V ויהיו היי G יהי יהי מרחקים: יהי
עץ יהי C_{\pi}=\{(\pi\left[v\right],v)\mid v\in V_{\pi}\setminus\{s\}\} וכן V_{\pi}=\{v\in V\mid \mathtt{BFS}\left(G,s\right).\pi\left[v\right]\neq \mathtt{Null}\}\cup\{s\} נגדיר s\in V יהי S\in V יהי יהי י
                                                                                                                                                .G_{\pi} = (V_{\pi}, E_{\pi})
                                                                                                                           טענה: יהי S \in V אזי גרף איר s \in V
```

- $\deg_{G_{\pi}}^{-}(s)=0$  מתקיים
- $\deg_{G_{\pi}}^{-}\left(v
  ight)=1$  מתקיים  $v\in V\left(G_{\pi}
  ight)$  •
- s,v בין ב־ $G_{\pi}$  בין מסלול בי  $v \in V\left(G_{\pi}\right)$  לכל
  - . הינו עץ $G_{\pi}$
- Gבין ביותר בין המסלול הקצר ביותר בין s,v בין בין  $G_\pi$  בין ויהי  $v\in V$  ביותר בין יהי  $v\in V$

מסלול אוילר: מסלול העובר על כל הקשתות בגרף.

מעגל אוילר: מסלול אוילר שהינו מעגל.

 $(\deg(v) \in \mathbb{N}_{\mathrm{even}}$  מתקיים  $v \in V$  מתקיים מעגל אוילר בGטענה: יהי G גרף קשיר ולא מכוון אזי (יש מעגל אוילר ב

אזי  $\deg\left(u
ight)\in\mathbb{N}_{\mathrm{even}}$  מתקיים מעגל אוילר: יהי G גרף קשיר ולא מכוון עבורו לכל

```
function EulerCircle(G, v):
    \sigma \leftarrow \text{List}(E(G))
    u \leftarrow Neighbor(v)
    while u \neq v do
        \sigma.append(\{v,u\})
         G = G \setminus \{\{v, u\}\}\
        u \leftarrow \text{Neighbor}(u)
    if length(\sigma) = |E(G)| then
      \mid return \sigma
    else
        w \leftarrow \{x \in V(G) \mid (\exists y \in V(G).(x,y) \in \sigma) \land (\deg(x) > 0)\}
        \sigma[w] = \text{EulerCircle}(G, w)
טענה: יהי v \in V(G) יהי ויהי ממן איים מתקיים עבורו לכל איי מתקיים ויהי עבורו לכל v \in V(G) איי היצה של מתקיים מתקיים איים מתקיים ויהי
                                                                                                                \mathcal{O}\left(|E|\right) הינה EulerCircle (G,v)
                                               . Neighbor (u) | \neq \varnothing פעילה מתקיים while כל עוד לולאת באלגוריתם EulerCircle טענה:
                . הינו מעגל אוילר. EulerCircle (G) אזי \deg (v) \in \mathbb{N}_{\mathrm{even}} מתקיים v \in V מתקיים הינו עבורו לכל
             .(\{v\in V(G)\mid \deg(v)\in \mathbb{N}_{\mathrm{odd}}\}=2) טענה: יהי G גרף קשיר ולא מכוון אזי (יש מסלול אוילר שאינו מעגל ב'
                        אזי\{v\in V\left(G
ight)\mid \deg\left(v
ight)\in\mathbb{N}_{\mathrm{odd}}\}|=2 אלגוריתם למציאת מסלול אוילר: יהי aגרף קשיר ולא מכוון עבורו
function EulerPath(G):
    \{v, u\} \leftarrow \{v \in V(G) \mid \deg(v) \in \mathbb{N}_{\text{odd}}\}
    G = G + \{\{v, u\}\}\
    \sigma = \operatorname{EulerCircle}(G, v)
    return \sigma \setminus \{v, u\}
                                                   .(לא קיים ב־G מעגל באורך אי־זוגיי) (דו־צדדיG איר מכוון אזי מכוון אזי G יהי G גרף אי
                                                                          אלגוריתם איהוי גרפים דו־צדדיים: יהי G גרף לא מכוון ופשוט אזי
function IsBipartite(G):
    (d, \pi, \operatorname{color}) \leftarrow \operatorname{BFS}(G)
    for (v,u) \in V do
     if d(v) = d(u) then return False
    end
    return True
                                                         .(IsBipartite (G) = \text{True}) איי (G דו־צדדי) מענה: יהי G גרף לא מכוון ופשוט אזי (G דו־צדדי)
      .|\sigma|=\min\{|	au|\ |t| מסלול קצר ביותר בין קודקודים: יהי S גרף ויהיו S גרף אזי מסלול \sigma מ־S ל־t עבורו T מסלול קצר ביותר בין קודקודים: יהי
                                                        גרף המסלולים הקצרים ביותר מקודקוד (גרף מק"ב): יהי G גרף ויהי s\in V נגדיר
                                                                      E'=\{e\in E\mid sאזי איזי איזי E'=\{e\in E\mid sאזי היוצא מ־פ
                                                   אזי אלגוריתם למציאת גרף המסלולים הקצרים ביותר מקודקוד: יהי G גרף ויהי אלגוריתם למציאת אר
```

```
function ShortestPathGraph(G, s):
     (d, \pi, \operatorname{color}) \leftarrow \operatorname{BFS}(G)
     E' \leftarrow E(G_{\pi})
     for (u,v) \in E(G) do
           if |\operatorname{height}_{G_\pi}(u) - \operatorname{height}_{G_\pi(v)}| = 1 then
            \mid E'.append((u,v))
     end
     return (V(G), E')
                                                         .(במק"ב) אזי פאר (G_\pi BFS טענה: תהא במק"ב) אזי מחברת בין מחברת פון מחברת פון אזי פון פאזי פון מענה:
                                                                    sב מ"ב מה"ב הינו גרף מק"ב מהינו אזי ShortestPathGraph (G,s) אזי s\in V מסקנה: יהי
                                                                           גריר המסלולים הקצרים ביותר בין קודקודים: יהיS,t\in V ויהיו גרף המסלולים הקצרים ביותר בין קודקודים:
                                                                               E' = \{e \in E \mid tאזי אזי E' = \{e \in E \mid tאזי אזי מ־סלול קצר ביותר היוצא מ־e\}
טענה: יהי S גרף מכוון ויהיו t בסיבוכיות אמן לחישוב גרף המסלולים הקצרים ביותר מיs ליל בסיבוכיות אמן ריצה אזי קיים אלגוריתם לחישוב און אזי מכוון ויהיו
                                                                                                                                                               \mathcal{O}(|V| + |E|)
                                                                                                                          אזי s \in V יהי גרף ויהי :DFS אלגוריתם
function DFS (G, s):
     (k,\pi) \leftarrow \operatorname{dict}(V)
     color \leftarrow dict(E)
     k[s] \leftarrow 1
     \pi[s] \leftarrow \text{Null}
     for u \in V \backslash s do
          k[u] \leftarrow 0
           \pi[u] \leftarrow \text{Null}
     end
     for e \in E do
      |\operatorname{color}[e] \leftarrow \operatorname{White}
     end
     i \leftarrow 2
     v \leftarrow s
     while (\exists u \in Adj(v).color[(v,u)] = White) \lor (\pi[v] \neq Null) do
           if \{u \in Adj(v) \mid \operatorname{color}[(v,u)] = \operatorname{White}\} \neq \emptyset then
                w \leftarrow \{u \in Adj(v) \mid \operatorname{color}[(v, u)] = \operatorname{White}\}\
                \operatorname{color}[(v, w)] \leftarrow \operatorname{Black}
                if k[w] = 0 then
                      k[w] \leftarrow i
```

```
O(|V|+|E|) סענה: יהי G גרף ויהי s\in V אזי סיבוכיות זמן הריצה של O(|V|+|E|) הינה O(|V|+|E|) אזי O(|V|+|E|) אוווון O(|V|+|E|) אווון O(|V|+|E|)
```

 $\pi[w] \leftarrow v$  $v \leftarrow w$  $i \leftarrow i + 1$ 

else

return  $(k,\pi)$ 

end

 $v \leftarrow \pi[v]$ 

אזי DFS יהי  $G_\pi$  אויהי יהי יהי יהי יער יהי יער אזי יער

- $.e\in E\left( G_{\pi}
  ight)$  עבורה  $e\in E\left( G
  ight)$  קשתות עץ: קשת •
- v שב של אב וכן  $u,v)\notin E\left(G_{\pi}\right)$  עבורה  $(u,v)\notin E\left(G\right)$  וכן הינו אב של •
- u שב של אב עכן וכן  $u,v) \notin E\left(G_{\pi}\right)$  עבורה עבורה  $(u,v) \in E\left(G\right)$  וכן הינו אב של
  - . שאינה קשת עץ או קדמית או שאינה  $e\in E\left(G\right)$  שאינה קשת  $\bullet$

 $G_{\pi}$  טענה: יהי  $G_{\pi}$  או  $G_{\pi}$  או עץ אזי u צאצא של ען אזי u בגרף או בגרף או בגרף כענה: יהי G

מסקנה: יהי G גרף לא מכוון אזי לא קיימות קשתות חוצות.

אלגוריתם DFS בעל זמני נסיגה: יהי ארף אזי

```
function DFS(G):
     (k, f, \pi, \text{color}, \text{low}) \leftarrow \text{dict}(V)
     for u \in V do
          k[u] \leftarrow 0
           \pi[u] \leftarrow \text{Null}
           color \leftarrow White
          low \leftarrow \infty
     end
     i \leftarrow 0
     for s \in V do
          if k[s] = 0 then
            DFS-VISIT(s, k, f, \pi, i)
     end
     return (k, f, \pi, low)
function DFS-VISIT(v, k, f, \pi, color, low, i):
     color[u] \leftarrow Gray
     i \leftarrow i + 1
     k[v] \leftarrow i
     for w \in Adj(v) do
          if (\operatorname{color}[v] = \operatorname{Gray}) \wedge (v \neq \pi[u]) then
           | low \leftarrow min(low[u], k[v])
           else if color[v] = White then
                \pi[w] \leftarrow v
                DFS-VISIT (w, k, f, \pi, \text{color}, \text{low}, i)
                low \leftarrow min(low[u], low[v])
     end
     color[u] \leftarrow Black
     i \leftarrow i + 1
     f[v] \leftarrow i
```

.DFS (G) אזי f אזי  $s \in V(G)$  ארף ויהי G גרף ויהי

 $v,u \in V$  איז ( $v,u \in V$  טענה: Gray Path Lemma: יהיע  $v,u \in V$  יהיע יהיע: יהיע

(f[v] < k[u])טענה: יהיו  $v,u \in V$  אזי אזי (u,v) קשת חוצה

משפט הסוגריים: יהי G גרף ויהיו אזי מתקיים בדיוק אחד מהבאים משפט משפט הסוגריים:

- $.G_{\pi}$  וכן אינם צאצא־אב ביער  $[k\left(u
  ight),f\left(u
  ight)]\cap [k\left(v
  ight),f\left(v
  ight)]=arnothing$  מתקיים  $\bullet$ 
  - $.G_{\pi}$  ביער ע צאצא u וכן ו $[k\left(u
    ight),f\left(u
    ight)]\subset[k\left(v
    ight),f\left(v
    ight)]$  פתקיים •
  - $.G_{\pi}$  מתקיים u צאצא v וכן ו $[k\left(u
    ight),f\left(u
    ight)]\supset[k\left(v
    ight),f\left(v
    ight)]$  פ

משפט המסלול הלבן: יהי G גרף ויהיו  $u,v\in V$  אזי ויהיו  $u,v\in V$  אזי ( $u,v\in V$  באלגוריתם ביער המסלול לבן: יהי  $u,v\in V$  אזי ( $u,v\in V$  יש מסלול לבן: יהי  $u,v\in V$  מ־ $u,v\in V$  מ־ $u,v\in V$ 

גרף מכוון אניקלי: גרף מכוון G בו לא קיים מעגל.

 $u\prec v$  אזי אוי  $(u,v)\in E$  אם  $u,v\in V$  אם לכל המקיים על סדר על אזי יחס אזי אזי ההי מיון טופולוגי: יהי

(G) משפט: יהי G גרף מכוון אזי (G אציקלי) אניקלי) משפט: יהי G גרף מכוון אזי

```
\mathcal{O}\left(|V|+|E|
ight) איז קיים אלגוריתם לבדיקת קיום מיון טופולוגי בסיבוכיות זמן ריצה G גרף מכוון אזי קיים אלגוריתם לבדיקת קיום מיון טופולוגי בסיבוכיות זמן ריצה
                                                                     (G^-משפט: יהי G גרף מכוון אזי G אציקלי) אציקלי) משפט: יהי G גרף מכוון אזי
                                            G טענה: יהי סופולוגי מהרצת DFS שרה מהרצת המתקבלי אזי אזי f המתקבלי אזי מיון טופולוגי על
                                                       \left. \left| G/_{\overrightarrow{G}} \right| < \left| G-\{v\}/_{\overrightarrow{G-\{v\}}} \right| עבורו v \in V\left(G\right) אז גרף מכוון אזי G אז מנתק: יהי G גרף מכוון ויהי v \in V אזי v \in V אזי v \in V קשת אחורית.
                                                                    .DFS (G) בהרצת low גרף אזי G ביותר: יהי G בהרצת המוקדם ביותר
                                                                      אלגוריתם למציאת כל הקודקודים המנתקים: יהיG גרף מכוון וקשיר אזי
function DetachableVertices(G):
    s \leftarrow V
     (k, f, \pi, \text{low}) \leftarrow \text{DFS}(G, s)
     A \leftarrow \operatorname{set}(V)
    if |Adj_{G_{\pi}}(s)| \neq 1 then
     A.append(s)
    for u \in V \backslash \{s\} do
         if \exists v \in children(u).low[v] \geq k[u] then
           A.append(u)
    end
    return A
                               \mathcal{O}(|V|+|E|) הינה Detachable Vertices (G) טענה: יהי G גרף מכוון וקשיר אזי סיבוכיות זמן הריצה של
                                         סענה: יהי G גרף מכוון וקשיר אזי Detachable Vertices (G) הינה קבוצת כל הקודקודים המנתקים.
וכן uכי מסלול מ־u גרף מכוון אזי קבוצה C\subseteq V מקסימלית בגודלה עבורה לכל u גרף מכוון אזי קבוצה אי מקסימלית בגודלה עבורה לכל
                                                                                                                                                  u^-ט ל־u^-
                                             G^T=(V,E') אזי אזי E'=\{(v,u)\mid (u,v)\in E\} אזי גרף מכוון נגדיר אזי יהי G אזי איי
                                                         (G^T) אזי (G^T) אזי (G^T) אזי איי רף מכוון ותהא אזי ((G^T) אזי ((G^T) אזי רף מכוון ותהא
                                                                אלגוריתם קוסראג'ו־שריר למציאת רכיבים קשירים היטב: יהי G גרף מכוון אזי
function SCC(G):
    (k, f, \pi) \leftarrow \text{DFS}(G)
     /* In the next line, in the main loop of DFS we go through the vertices in decreasing order of f[u]
                                                                                                                                                     */
    (k', f', \pi') \leftarrow \text{DFS}(G^T)
    A \leftarrow \operatorname{set}(\operatorname{set}(V))
    for v \in V do
         A.append \left( [v]_{\overrightarrow{G_{\pi}^T}} \right)
    end
    return A
   .G^{*}=\left(\operatorname{SCC}\left(G
ight),E^{*}
ight) אזי E^{*}=\left\{ \left(A,B
ight)\in\operatorname{SCC}\left(G
ight)^{2}\mid\exists u\in A.\exists v\in B.\left(u,v
ight)\in E
ight\} אזי היי G גרף הרכיבים: יהי
                                                                                            אלגוריתם למציאת גרף הרכיבים: יהי G גרף מכוון אזי
```

```
 \begin{aligned} &(u,v) \in L \text{ do} \\ &\text{if } [v] \xrightarrow[G_{\pi}^T]{} \neq [u] \xrightarrow[G_{\pi}^T]{} \text{ then} \\ & \bigg| E^*.\text{append} \left( \left( [v] \xrightarrow[G_{\pi}^T]{}, [u] \xrightarrow[G_{\pi}^T]{} \right) \right) \end{aligned} 
     end
     return (V^*, E^*)
                                                                                                                    למה: יהי G גרף מכוון אזי G^st אציקלי.
                                                                                                                      אזי U\subseteq V אזי גרף ותהא G יהי
                                                                                                               .k\left( U
ight) =\min_{u\in U}\left( k\left[ u
ight] 
ight) זמן גילוי: •
                                                                                                            f(U) = \max_{u \in U} (f[u]) זמן נסיגה: •
                                         f\left(C_{2}
ight) < f\left(C_{1}
ight) אזי אזי \left(C_{1},C_{2}
ight) \in E\left(G^{st}
ight) באשר רק"ה באשר רק"ה מכוון יהיו היי G גרף מכוון יהיו
                               f\left(C_{2}
ight)>f\left(C_{1}
ight) אזי אזי איזי \left(C_{1},C_{2}
ight)\in E\left(\left(G^{T}
ight)^{*}
ight) באשר באשר רק"ה באשר מסקנה: יהי G גרף מכוון יהיו
                                                                          (C\in \operatorname{SCC}(G))אזי (C\cap G)הא(C\cap G)אזי ויהי (C\cap G)אזי משפט: יהי (C\cap G)אזי ויהי
                                                                                               G^* = \text{KosarajuSharir}(G) אזי גרף מכוון אזי G גרף מכוון אזי
                                                                   \exists v \in V. \exists s \in S. s 	o v המקיימת S \subseteq V הכוון אזי גרף מכוון אזי יהי
                                                                                     אלגוריתם למציאת קבוצת מוצא מינימלית: יהי G גרף מכוון אזי
function MinimalOriginSet(G):
    A \leftarrow \operatorname{set}(V(G))
     G^* \leftarrow \text{ComponentGraph}(G)
     for C \in V(G^*) do
         v \leftarrow \{u \in C \mid \nexists w \in V(G) \backslash C.(w,u) \in E(G)\}
          A.append(v)
     end
     return A
                                                                        . קבוצת מוצא מינימלית MinimalOriginSet (G) אזי מכוון אזי G יהי הי G גרף מכוון אזי
                                   \mathcal{O}\left(|V|+|E|
ight) הינה MinimalOriginSet (G) טענה: יהי G גרף מכוון וקשיר אזי סיבוכיות זמן הריצה של
\mathcal{O}\left(|V|+|E|
ight) איז קיים אלגוריתם הבודק האם קיים הילוך \sigma העובר על S\subseteq V איז קיים אלגוריתם הבודק האם \sigma
                                                                                              (G,w) אזי w:E	o\mathbb{R} גרף ממושקל: יהי
                                                V\left(T
ight)=V\left(G
ight) עץ וכן T באשר באשר אי תת־גרף אזי תת־גרף קשיר לא מכוון אי תת־גרף T\leq G באשר
                                           .w\left(T
ight) = \sum_{e \in E\left(T
ight)} w\left(e
ight) אזי פורש עץ פורש משקל אמכוון ויהי קשיר א מכוון אזי Gיהי יהי משקל עץ פורש: יהי
      .w\left(T
ight)=\min\left\{ w\left(S
ight)\mid G עץ פורש מינימלי (עפ"מ): יהי G גרף קשיר לא מכוון אזי עץ פורש T\leq G עבורו אזי עץ פורש זהי G יהי
                                                                                      A \uplus B = V(G) עבורם A, B \subseteq V(G) אזי אזי G יהי יהי
                                \{(u,v)\in E\left(G\right)\mid (u\in A)\land (v\in B)\} חתך אזי A,B\subseteq V\left(G\right) יהי גרף ויהי G גרף יהי
                                                           . בעל מעגל יחיד T+\{e\} אזי e\in E\left(G\right)ackslash E\left(T
ight) בעל מעגל יחיד יהי T\leq G יחיד.
עץ T+\{e_1\}-\{e_2\} עץ פורש תהא עץ פורש e_2\in E (T+\{e_1\}) ותהא ותהא ותהא e_1\in E (G)\setminus E עץ פורש תהא עץ פורש תהא
                                                                                                                                                               פורש.
                                                           . טענה: יער בעל שני עצים T-\{e\} אזי אי פורש ותהא פורש פורש פורש דער אזי T \leq G יהינו עצים.
                      [v]_{\overbrace{T-\{e\}}},V\left(G
ight)\setminus\left[v\right]_{\overbrace{T-\{e\}}} אזי v\in V\left(G
ight) ויהי e\in E\left(T
ight) חתך של T\leq G מסקנה: יהי T\leq G
                                                          אלגוריתם גנרי למציאת עץ פורש מינימלי: יהי G גרף קשיר לא מכוון וממושקל w אזי
```

function KosarajuSharir(G):

 $V^* \leftarrow \operatorname{SCC}(G)$   $E^* \leftarrow \operatorname{set}((V^*)^2)$ for  $(u, v) \in E$  do

```
function MST(G, w):
       color \leftarrow dict(E)
       for e \in E do
        |\operatorname{color}[e]| = \operatorname{White}
       end
       while \exists e \in E.color[e] = White do
             \mathsf{Blueless} \leftarrow \{A \subseteq V \mid \forall e \in (A^2 \cap E).\mathsf{color}[e] \neq \mathsf{Blue}\}
             Redless \leftarrow \{ \sigma \text{ circle in } G \mid \forall i \in [\text{len}(\sigma)].\text{color}[(\sigma[i], \sigma[i+1])] \neq \text{Red} \}
             if Blueless \neq \emptyset then
                    A \leftarrow \text{Blueless}
                    f \leftarrow \operatorname{argmin}_{e \in A^2 \cap E}(w(e))
                    color[f] = Blue
             if Redless \neq \emptyset then
                    \sigma \leftarrow \text{Redless}
                     f \leftarrow \operatorname{argmax}_{e \in \sigma}(w(e))
                    color[f] = Red
       end
      return (V, \{e \in E \mid \operatorname{color}[e] = \operatorname{Blue}\})
```

 $e \in E$  אזי קיימת MST (G) באיטרציה של color [a]= White עבורה ותהא  $a \in E$  אוי ותהא w ותהא אזי קיימת אויר ניתוח לארגעה

. אות אובעת |E| צובעת MST (G) אזי w אווע מסקנה: יהי G גרף אור קשיר איז מסקנה:

עפ"מ עבורו אזי פענה: איטרציה אזי אזי מכוון וממושקל אזי אזי אזי אזי אזי אזי עפ"מ עבורו אזי ארף אזי ארף אזי יהי עבורו עפ"מ עבורו אזי מכוון וממושקל אזי אזי בכל איטרציה אזי איי עפ"מ עבורו

- $.e\in E\left( T
  ight)$  מתקיים color  $\left[ e
  ight] =$  Blue המקיימת  $e\in E$  לכל
- $.e \notin E\left(T
  ight)$  מתקיים color  $[e]=\mathrm{Red}$  המקיימת  $e \in E$  לכל

G עפ"מ של MST G עפ"מ אזי w אוי מסקנה: יהי G גרף עפ"מ אל עפ"מ ער מסקנה: יהי

אזי w אזי וממושקל אזי אינימלי: יהי G אזי פורש מינימלי: אזי אלגוריתם פרים למציאת אי

```
function Prim's Algorithm (G):
```

```
color \leftarrow dict(E)
U \leftarrow \operatorname{set}(V)
for e \in E do
     color[e] = White
end
r \leftarrow V
U.append(r)
while U \neq V do
      (u,v) \leftarrow \operatorname{argmin}_{e \in U \times (V \setminus U)}(w(e))
     color[(u, v)] = Blue
     U.append(v)
     for w \in U do
           if (w,v) \in E then
             |\operatorname{color}[(w,v)]| = \operatorname{Red}
     end
end
return (V, \{e \in E \mid \operatorname{color}[e] = \operatorname{Blue}\})
```

. נעשית כמו באלגוריתם אזי כל צביעת קשת באלגוריתם Prim'sAlgorithm (G) טענה: יהי G גרף קשיר לא מכוון וממושקל w אזי כל צביעת קשת באלגוריתם G עפ"מ של Prim'sAlgorithm (G) אזי w אוון w אזי w אוון w אוון w אזי w אוון w א

משפט: עם ערימת מינימום בסיבוכיות אז עם Prim'sAlgorithm (G) אזי ניתן לממש אזי ניתן ניתן אזי עס ערימת מינימום אזי יהי G יהי G יהי G יהי G יהי G יהי G יהי לממש אזי ניתן לממש אוני ניתן לממש אזי ניתן לממש אזי ניתן לממש אוני ניתן לממש

 $\mathcal{O}\left(|E|+|V|\log|V|\right)$  בסיבוכיות זמן ריצה Prim'sAlgorithm (G) אזי ניתן לממש אזי ניתן לממש אזי ניתן לממש אזי ניתן למציאת עץ פורש מינימלי: יהי G גרף קשיר לא מכוון וממושקל אזי w אלגוריתם קרוסקל למציאת עץ פורש מינימלי: יהי

```
\begin{array}{l} \text{function Kruskal'sAlgorithm}(G) \text{:} \\ & \operatorname{color} \leftarrow \operatorname{dict}(E) \\ & L \leftarrow \operatorname{sort}(E) \\ & \text{for } (u,v) \in L \text{ do} \\ & | & \operatorname{if} \exists \sigma \in \{u \rightarrow v\}. \forall i \in [n]. \operatorname{color}(\sigma(i)) = \operatorname{Blue \ then} \\ & | & \operatorname{color}[e] = \operatorname{Red} \\ & | & \operatorname{color}[e] = \operatorname{Blue} \\ & | & \operatorname{end} \\ & | & \operatorname{return} \ (V, \{e \in E \mid \operatorname{color}[e] = \operatorname{Blue}\}) \end{array}
```

. נעשית כמו באלגוריתם אזי כל צביעת קשת באלגוריתם אזי כמו באלגוריתם נשית כמו באלגוריתם הגנרי. אזי כמו באלגוריתם הגנרי. אזי מסקנה: יהי G גרף קשיר לא מכוון וממושקל w אזי G אזי G גרף קשיר לא מכוון וממושקל אזי G אזי אזי G גרף קשיר לא מכוון וממושקל אזי G

עם Union-Find עם Kruskal'sAlgorithm (G) אי איי ניתן לממש איי ניתן עם איי ניתן עם איי ניתן איי ניתן עם עם U גרף קשיר לא מכוון וממושקל w איי ניתן לממש איי ניתן U וכן סיבוכיות אמן ווממושקל  $\mathcal{O}\left(|E|\log|V|\right)$ .

אזי שח"ע אזי ש באשר א באשר ש ארגוריתם Borůvska מינימלי: יהי מינימלי: יהי ש מינימלי באשר ש ארגוריתם ארגוריתם אווי שווי מינימלי

## function Borůvska's Algorithm (G):

```
\begin{array}{l} \operatorname{Trees} \leftarrow \operatorname{set}(\operatorname{set}(G)) \\ \text{for } v \in V \text{ do} \\ | \operatorname{Trees.append}(\{v\}) \\ \text{end} \\ \text{while } |Trees| \neq 1 \text{ do} \\ | \operatorname{for } T \in Tree \text{ do} \\ | (u,v) \leftarrow \operatorname{argmin}_{(u,v) \in V(T) \times V(G)}(w((u,v))) \\ | S \leftarrow \{S \in Tree \mid u \in V(S)\} \\ | S \leftarrow S + T + \{(u,v)\} \\ | \operatorname{Trees.Remove}(T) \\ | \text{end} \\ \text{end} \\ A \leftarrow \operatorname{Trees} \\ \text{return } A \end{array}
```

 $\mathcal{O}\left(|E|\log|V|\right)$  הינה Borůvska's Algorithm (G) אזי סיבוכיות און חח"ע איז חח"ע באשר ש הינה מכוון וממושקל ש באשר ש חח"ע איז קיים ויחיד T < G עפ"מ.

G עפ"מ של Borůvska's Algorithm (G) עפ"מ של באשר w באשר ש מכוון וממושקל אזי ההי משפט: יהי

 $T \leq G$  משפט: יהי G גרף קשיר לא מכוון וממושקל w תהא ערהא יהי  $A \subseteq E$  יהי  $A \subseteq E$  משפט: יהי  $A \subseteq E$  אזי קיים עפ"מ  $e \notin E(T)$  וכן  $A \subseteq E(T)$  עבורו

 $lpha_i=eta_i$  וכן n=m וכן אזי הקשתות כולל כפילויות משקליי ה $lpha_1\leq\ldots\leqeta_m$  ו־מענה: יהיו עפ"מ ויהיו עפ"מ ויהיו  $lpha_1\leq\ldots\leqlpha_n$  וכן הייו לכל וכל לכל ויהיו הייו הייו משקליי הייו משקליי ויהיו משקליי הייו משקליי הייו משקליי ויהיו משקליי הייו משקליי ויהיו משקליי הייו משקליי הייו משקליי ויהיו משקליי הייו משקליי ויהיו משקליי הייו משקליי הייו משקליי ויהיו משקליי הייו משקליי הייו משקליי הייו משקליי הייו משקליי ויהייו משקליי הייו משקליי ויהייו משקליי הייו משקליי הייו

w אלגוריתם למציאת עץ פורש מינימלי המכיל מספר מקסימלי של קשתות מקבוצה מסוימת: יהי G גרף קשיר לא מכוון וממושקל ותהא  $F\subseteq E$  אזי

```
function PrioritizeMST(G, w, F):
     \omega \leftarrow (V(G) \rightarrow \mathbb{R})
     m \leftarrow \min(\{|w(e_1) - w(e_2)| \mid (e_1, e_2 \in E) \land (w(e_1) \neq w(e_2))\}
     for e \in E do
         if e \in F then
          \omega(e) \leftarrow w(e)
         else
           | \omega(e) \leftarrow w(e) + \varepsilon
     end
     return Kruskal'sAlgorithm(G, \omega)
                                           w'טענה: תהא T עפ"מ ביחס ל־w PrioritizeMST אזי עפ"מ ביחס ל־T עפ"מ ביחס ל־T עפ"מ ביחס ל
                                                                          wעפ"מ ב־G ביחס עפ"מ ב־G אזי PrioritizeMST (G,w) אזי אזי די מסקנה: תהא
                                                        אזי i \in [n] לכל s_i < f_i באשר בעיית שיבוץ המשימות: יהיו יהיו s_i < f_i באשר בעיית
                                                                                   \max\left\{|A|\mid (A\subseteq\left\{[s_1,f_i]\right\}_{i=1}^n)\wedge (\forall I,J\in A.I\cap J=\varnothing)\right\}
                                  אזי i \in [n] אזי אינוריתם חמדן לבעיית שיבוץ המשימות: יהיו יהיו s_i < f_i באשר באשר אזי אלגוריתם אזי
function ActivitySelectionProblem(s_1, \ldots, s_n, f_1, \ldots, f_n):
     F \leftarrow \operatorname{list}([s_1, f_1], \dots, [s_n, f_n])
     /* In the next line, we put all the parameters in F and sort it based on f_i
     F \leftarrow \operatorname{sort}(\{f_1, \dots, f_n\})
     X \leftarrow \operatorname{list}([s_1, f_1], \dots, [s_n, f_n])
     X \leftarrow \varnothing
     for k \in [1, \ldots, n] do
```

 $\mathcal{O}(n\log(n))$  הינה ActivitySelectionProblem אי סיבוכיות זמן הריצה של  $s_i < f_i$  באשר הינה  $s_1 \ldots s_n, f_1 \ldots f_n \in \mathbb{R}$  טענה: יהיו

 $.\ell(\sigma) = \min\{\ell(\tau) \mid au \in \{s o t\}\}$  מסלול קצר ביותר בין קודקודים: יהי G גרף ממושקל  $\ell$  ויהיו  $s,t \in V$  אזי מסלול  $\sigma$ למה: יהיו t קשת השייכת למעגל שלילי אזי קיים מסלול לt וכן כל מסלול מ־t לא עובר דרך קשת השייכת למעגל שלילי אזי קיים מסלול למה:

למה: יהיו sעבורם קיים מסלול מיsלי לכן קיים מסלול מיsל לכל אזי לא קיים מסלול מיsלמה: יהיו

sבעיית המסלולים הקצרים מנקודת מוצא (SSSP): יהי S גרף מכוון ממושקל  $s\in V$  אזי איי  $T\leq G$  עץ פורש בו כל מסלול מ

עבורו  $X^*$  באיטרציה פתרון לבעיה אכניים ActivitySelectionProblem משפט: לכל באיטרציה ה־ $k \in [n]$ 

. מסקנה: יהיו איבוץ בעיית שיבוץ באשר אזי אזי אוא איז אוא איז המשימות. באשר  $s_1 \dots s_n, f_1 \dots f_n \in \mathbb{R}$  מסקנה: יהיו  $.\ell=1$  בורו גרף הוא  $\ell$  הכוונה היא "אורך הקשת" ובכך אנו "מכלילים" גרף ללא משקל בתור גרף עבורו הקשת" ובכך אנו

רביותר.  $\sigma[i],\ldots\sigma[i+k]$  אזי  $\sigma[i+k]$  מסלול קצר ביותר ויהי מסלול קצר ביותר ויהי אזי l אוי וממושקל  $\ell$  ויהי ומכוון וממושקל בלגוריתם בלמן-פורד מדיים ביותר מקדים ביותר מנקודת אלגוריתם בלמן-פורד למציאת מסלולים קצרים ביותר מנקודת מוצא: יהי

if  $X = \emptyset$  then X.append(L[k])

end return X

X.append(L[k])

else if  $L[k] \cap X$ .last =  $\emptyset$  then

\*/

 $([s_i, f_i] \in X^*) \iff ([s_i, f_i] \in X)$ 

tל־s פשוט קצר ביותר בין

tל־ל s פשוט קצר ביותר בין

Gל־ט הינו מסלול קצר ביותר ב־v

 $.\delta\left(s,t
ight)=\inf_{\sigma\in\left\{s o t
ight\}}\ell\left(\sigma
ight)$  אזי  $s,t\in V$  ויהיו אויהים גרף ממושקל G יהי

 $\delta\left(u,v
ight) \leq \delta\left(u,w
ight) + \delta\left(w,v
ight)$  אזי  $u,v,w \in V$  למה אי־שיוויון המשולש: יהיו

```
d[s] \leftarrow 0
    for u \in V do
         d[u] \leftarrow \infty
         \pi[u] \leftarrow \text{None}
    end
     (c,i) \leftarrow 1
    while (i \leq |V|) \land (c \neq 0) do
         for (u,v) \in E do
          c \leftarrow \text{Relax}(\ell, d, u, v)
         end
    end
    return c
function Relax (\ell, d, u, v):
    if d[v] > d[u] + \ell(u, v) then
         d[v] \leftarrow d[u] + \ell(u, v)
         \pi[v] \leftarrow u
    return 1 return 0
       \delta(s,v) \leq d\left[u
ight] + \ell\left((u,v)
ight) אזי אזי אונן BellmanFord מתקיים אזי וכן בריצת s,u,v \in V למה: יהיו
מסקנה: יהיו \delta(s,v) \leq d[v] איי לאחר הרצת BellmanFord מסקנה: יהיו \delta(s,v) \leq d[v] באשר הרצת הרצת וען בריצת בריצת וען בריצת
                                                                                                            \delta(s,v) < d[v] מתקיים Relax (u,v)
נקבל כי לכל Relax עבורו לכל איז אי אי אי מתקיים BellmanFord בריצת איי עבורו לכל s\in V איי איי למה: יהי א s\in V למה:
                                                                                                                  \delta\left(s,v\right)\leq d\left[v\right] מתקיים v\in V
        d\left[v
ight]=\infty מסקנה: יהיו Relax מתקיים BellmanFord מתקיים מחקיים אזי לאחר כל רצף פעולות איז פביצת או מסקנה: יהיו
d\left[v
ight]=\delta\left(s,v
ight) בקבל כי Relax מסקנה: יהיו איי לאחר כל רצף מתקיים BellmanFord מתקיים אויי לאחר כל רצף פעולות
                 מתקיים הפעלת אזי לאחר הפעלת מחקיים הפול מתקיים מחקיים BellmanFord מתקיים אזי לאחר מינו למה: יהיו s,t\in V
                                                                 d\left[t
ight] \leq \ell\left(\sigma
ight) נקבל כי (Relax \left(\sigma\left[0
ight],\sigma\left[1
ight]
ight),\ldots, Relax \left(\sigma\left[n-1
ight],\sigma\left[n
ight]
ight)
```

function BellmanFord( $G, \ell, s$ ):

 $(d,\pi) \leftarrow \operatorname{dict}(V)$ 

מסקנה: יהי $s \in V$  אזי

מחזיר 1.

- (sיים מעגל שלילי אשר ניתן להגיע אליו מ־) BellmanFord) •
- .( $d\left[v
  ight]=\delta\left(s,v
  ight)$  מתקיים (לא קיים מעגל שלילי אשר ניתן להגיע אליו מ־s וכן לכל (א קיים מעגל שלילי אשר ניתן להגיע אליו מ־s וכן (א קיים מעגל שלילי אשר ניתן להגיע אליו מר $v\in V_{\pi}\setminus\{s\}$  וכן (א נגדיר  $v=v\in V_{\pi}\setminus\{s\}$  וכן v=v=v וכן (א נגדיר אזי v=v=v וכן (א נגדיר אזי (א נגדיר לא קיים מעגל שלילי אשר ניתן להגיע אליו מרv=v=v וכן (א נגדיר לא קיים מעגל שלילי אשר ניתן להגיע אליו מרv=v=v וכן (א נגדיר לא קיים מעגל שלילי אשר ניתן להגיע אליו מרv=v=v וכן (א קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן להגיע אליו מרv=v=v וכן (א קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן להגיע אליו מרv=v=v=v וכן (א קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן להגיע אליו מרv=v=v=v=v=v וכן (א קיים מעגל שלילי אשר ניתן לא קיים מעגל שלילי אשר ניתן לא ני

וכן  $s\in V$  ובאית כאשר אפיים מעגל שלילי אשר ניתן להגיע אליו מ־s אזי BellmanFord יוצא מהלולאה הראשית כאשר אריים מעגל שלילי אשר ניתן להגיע אליו מ־s

וכן i=|V| יוצא מהלולאה הראשית כאשר i=|V| אשר ניתן להגיע אליו מיז אוי פווא עבורו קיים מעגל שלילי שר ניתן להגיע אליו מיז איז איז איז איז איז שלילי שלילי שלילי אשר ניתן להגיע אליו מיז איזי איז איז פון ווכן  $s\in V$ 

. אזי BellmanFord אזי BellmanFord באיזשהו שלב של בעץ מעגל בעץ אזי מעגל מעגל בעץ אזי ויהי אזי אזי ויהי אזי אזי ויהי אזי אזי פו

מכיל מעגל שלילי. BellmanFord אזי עץ אליו להגיע אשר ניתן שלילי שלילי שלילי שלילי עבורו קיים מעגל שלילי אשר ניתן להגיע אליו מי

.SSSP מסקנה: יהי BellmanFord אזי  $s \in V$  מסקנה: יהי

 $d\left[v
ight]=\delta\left(s,v
ight)$  מתקיים  $v\in V$  מחזיר 10 וכן לכל

 $\mathcal{O}\left(|E|\cdot|V|
ight)$  אזי אפנט: פעל סיבוכיות בעל BellmanFord משפט: יהי א  $s\in V$  משפט

הערה: נניח כי SSSP אזי קיים אלגוריתם אזי קיים וכן  $\ell:E \to \mathbb{Z}$  בסיבוכיות ממן ריצה  $\ell(e) \geq -W$  וכן וכן  $\ell:E \to \mathbb{Z}$  בסיבוכיות מון ריצה . $\mathcal{O}\left(|E|\log^2(|V|)\log(|V|\cdot W)\log\log(|V|)\right)$ 

אלגוריתם לבדיקת קיום מעגל במשקל 0 בגרף מכוון חסר מעגלים שליליים: יהי G יהי אלגוריתם במשקל ס בגרף מכוון אלי

```
\begin{array}{c|c} \text{function IsZeroCircle}(G,\ell) \text{:} \\ V \leftarrow V \uplus \{s\} \\ \text{for } v \in V \backslash \{s\} \text{ do} \\ & E \leftarrow E \cup \{(s,v)\} \\ & \ell((s,v)) \leftarrow 0 \\ \text{end} \\ & (c,d,\pi) \ gets \text{BellmanFord}(G,\ell,s) \\ \text{for } e \in E \text{ do} \\ & \text{if } d(v) \neq d(u) + \delta(u,v) \text{ then} \\ & \mid E \leftarrow E \backslash \{(s,v)\} \\ \text{end} \\ & \text{if } \exists \text{ circle } C \in G \text{ then return True} \\ & \text{return False} \end{array}
```

טענה: בריצת IsZeroCircle אחר מחיקת כל הקשתות נקבל את גרף מק"ב מC אזי C וואסר מחיקת כל הקשתות קיים מעגל C אזי IsZeroCircle אחר מחיקת כל הקשתות קיים מעגל C אזי וואסענה: יהי C מעגל עבורו C אזי בריצת אזי בריצת וואסענה: יהי C מעגל עבורו C אזי בריצת אזי בריצת וואסר מעגל ממשקל C בגרף מחזיר IsZeroCircle) וואסר מעגלים שליליים אזי C בעל מעגל ממשקל C הינה C וואסר מעגלים שליליים אזי סיבוכיות זמן הריצה של IsZeroCircle הינה C מכוון חסר מעגלים שליליים אזי סיבוכיות מון אביקלי: יהי C מכוון אציקלי ויהי C אזי אלגוריתם למציאת מסלולים קצרים ביותר מנקודת מוצא בגרף מכוון אציקלי: יהי C מכוון אציקלי ויהי C

```
function SSSP-DAG(G, \ell, s):
    (d,\pi) \leftarrow \operatorname{dict}(V)
    d[s] \leftarrow 0
    for u \in V do
         d[u] \leftarrow \infty
         \pi[u] \leftarrow \text{None}
    end
    /* Knuth'sAlgorithm is an algorithm to compute a topological sorting.
                                                                                                                                                        */
    f \leftarrow \text{Knuth'sAlgorithm}(G)
    for i \in [1, \ldots, |V|] do
         for v \in Adj(f(i)) do
          Relax((f(i), v))
         end
    end
    return (d,\pi)
```

טענה: יהי G מכוון אציקלי ויהי  $s\in V$  אזי SSSP-DAG (G) פתרון לבעיית SSSP-DAG (G) אזי סענה: יהי G מכוון אציקלי ויהי  $s\in V$  אזי סיבוכיות זמן הריצה של SSSP-DAG (G) הינה (|E|+|V|). אלגוריתם דייקסטרה למציאת מסלולים קצרים ביותר מנקודת מוצא בגרף ללא משקלים שליליים: יהי G גרף מכוון עבורו  $\ell\geq 0$  ויהי S אזי

```
d[s] \leftarrow 0
    for u \in V do
         d[u] \leftarrow \infty
         \pi[u] \leftarrow \text{None}
    end
     Q.insert((s, d[s]))
    while Q \neq \varnothing do
         u \leftarrow Q.\min
         for v \in Adj(u) do
             if d[v] = \infty then
                  \pi[v] \leftarrow u
                  d[v] \leftarrow d[u] + \ell(u, v)
                  Q.insert((v, d[v]))
             else if d[u] + \ell(u, v) < d[v] then
                  \pi[v] \leftarrow u
                  d[v] \leftarrow d[u] + \ell(u, v)
                  Q.decrease-key((v, d[v]))
         end
    end
    return (d,\pi)
                                                  d\left[u
ight]=\delta\left(s,u
ight) אזי מרקה מ־Dijkstra למה: יהיו אזי אזי אוי אזי אזי מחקה בריצת למה: יהיו
                                                                             \ell \geq 0 כאשר SSSP משפט: יהי היי אוי משפט: אזי אזי משפט משפט יהי
                  \mathcal{O}\left(|E|+|V|\cdot\log\left(|V|
ight)
ight) איז ניתן לממש את Dijkstra אי בסיבוכיות אמן איז ניתן לממש את s \in V משפט: יהי
D_{u,v}=\delta\left(u,v
ight) מתקיים u,v\in V עבורו לכל D\in M_{|V|}\left(\mathbb{R}
ight) אזי אזי B גרף מכוון וממושקל אזי מרון וממושקל אזי מתקיים אזי מתקיים מכוון וממושקל אזי
                                  \Pi(\Pi_{u,v},v)\in\sigma מ־ע ל־\sigma מ"ע מ\sigma מ"ע מיע מסלול קצר א קיים מסלול קצר ביותר \Pi\in M_{|V|}(V) וכן
                                                                                                p:V	o\mathbb{R} אזי גרף אזי G יהי פוטנציאל: יהי
מתקיים (u,v)\in E מתקיימים u,v\in V מונקציית משקל מונקציית פונקציית פונקציית פונקציית פונקציית משקל מותאמת: תהא
                                                                                                       \ell_p((u,v)) = \ell((u,v)) + p(u) - p(v)
                          \ell_p(\sigma)=\ell\left(\sigma
ight)+p\left(s
ight)-p\left(t
ight) אזי ל־ז אזי s,t\in V משפט: תהא g פונקציית פוטנציאל יהיו
(מסלול קצר ביותר ביחס ל־t מסלול מיs ויהי מסקנה: s ויהי היו וויהי s פונקציית פוטנציאל היו וויהי s מסלול מt
                                                                                                                                  ביותר ביחס ל־(\ell_p).
                                                                       \ell_p\left(\sigma
ight)=\ell\left(\sigma
ight) מסקנה: תהא מונקציית פוטנציאל ויהי מסקנה פונקציית פונקציית פוטנציאל ויהי
                                            .\delta_{\ell}\left(s,t
ight)=\delta_{\ell_{p}}\left(s,t
ight)-p\left(s
ight)+p\left(t
ight) אזי s,t\in V מסקנה: תהא p פונקציית פוטנציאל ויהיו
                                    \ell_p \geq 0 בורה p עבורה פוטנציאל פיזבילית: יהי p גרף מכוון וממושקל \ell אזי פונקציית פוטנציאל פיזבילית: יהי
          משפט: יהי G גרף מכוון וממושקל \emptyset אזי (קיימת פונקציית פוטנציאל פיזבילית)(G) מצוייד עם \emptyset חסר מעגלים שליליים).
```

אלגוריתם למציאת פונקציית פוטנאציל פיזבילית: יהי G גרף מכוון וממושקל אזי

function Dijkstra $(G, \ell, s)$ :  $\begin{array}{c|c} Q \leftarrow \text{heap}((V, \text{int})) \\ (d, \pi) \leftarrow \text{dict}(V) \end{array}$ 

```
function FeasiblePotential(G, \ell):
         G' \leftarrow G \uplus \{s\}
         for v \in V(G) do
                  E(G') \leftarrow E(G') \cup \{(s,v)\}
                 \ell((s,v)) \leftarrow 0
         end
         c \leftarrow \text{BelmanFord}(G', \ell, s)
         if c=1 then return None
         p \leftarrow (V(G) \rightarrow \mathbb{R})
         for v \in V(G) do
           p(v) \leftarrow \delta(s, v)
         end
         return p
                                                                                                                                                                                                               טענה: יהי G גרף מכוון וממושקל \ell אזי
                                                                                                             .(None מחזיר FeasiblePotential (G,\ell)) בעל מעגל שלילי בעל מעגל מעגל \ell מצוייד עם G
                   פחזיר פונקציית פוטנציאל פיזבילית). FeasiblePotential (G,\ell) פיזבילית פוטנציאל פיזבילית פוטנציאל פיזבילית). \ell
                                                                                                                 אלגוריתם ג'ונסון לבעיית כל המסלולים הקצרים: יהי G גרף מכוון וממושקל אזי
function Johnson (G, \ell):
         p \leftarrow \text{FeasiblePotential}(G, \ell)
         if p = None then return None
         \ell_p \leftarrow (E \to \mathbb{R})
         for (u,v) \in E do
           \ell_p((u,v)) = \ell((u,v)) + p(u) - p(v)
         end
          (D_{\ell_p}, D_{\ell}) \leftarrow M_{|V|}(\mathbb{R})
         \Pi \leftarrow M_{|V|}(E)
         for v \in V do
                  (d,\pi) \leftarrow \text{Dijkstra}(G,\ell_p,v)
                  /* Here D and \Pi will be simplified, to get a solution to APSP as needed we can modify a bit Dijkstra's
                          algorithm to calculate D and \Pi on the way to get constant time for this assignment.
                  D_v \leftarrow d
                \Pi_v \leftarrow \pi
         end
         for (u,v) \in E do
                  D_{\ell}((u,v)) = D_{\ell_p}((u,v)) - p(u) + p(v)
         return (D,\Pi)
                                                                                                                           .
APSP משפט: יהי לבעיית Johnson (G,\ell) אזי שוח וממושקל לבעיית משפט: יהי לבעיית מחון וממושקל א
                              \mathcal{O}\left(|E|\,|V|+|V|^2\log\left(|V|
ight)
ight) הינה הינה אינ סיבוכיות אמן הריצה אז סיבוכיות הינה אזי סיבוכיות אזי סיבוכיות הינה אזי סיבוכיות אוני סיבוכיות אוני סיבוכיות אוני סיבוכיות אונים שליבונית אונים אונים שליבונית שליבונית אונים שליבונית אונים שליבונית אונים שליבונית שליב
                                                                                            A st B \in M_{m 	imes k} \left( \mathbb{F} 
ight) אזי B \in M_{n 	imes k} \left( \mathbb{F} 
ight) ותהא A \in M_{m 	imes n} \left( \mathbb{F} 
ight) תהא :Min Plus מכפלת
                                                                                                                                                                                                 (A*B)_{i,j} = \min_{k=1}^{n} (A_{i,k} + B_{k,j}) באשר
                                                                                                              \mathcal{O}\left(n^3
ight) אינ סיבוכיות אמן הריצה של A*B\in M_n\left(\mathbb{F}
ight) הינה טענה: תהיינה
                                                                                                                                          A*B*C=(A*B)*C אזי A,B,C\in M_n(\mathbb{F})טענה: תהיינה
                                                                                                              .\delta_k(s,v) = \min\{\ell(\sigma) \mid (\sigma \in \{s \to v\}) \land (|\sigma| \le k)\} אזי s,v \in V סימון: יהיו
                                                                                                                                     .\delta_{k}\left(s,v
ight)=\min_{u\in V}\left(\delta_{k-1}\left(s,v
ight)+\ell\left(u,v
ight)
ight) אזי s,v\in V טענה: יהיו
                                                                                                    v\in V לכל (\delta_{k}\left(s
ight))_{v}=\delta_{k}\left(s,v
ight) באשר \delta_{k}\left(s
ight)\in M_{1	imes |V|}\left(\mathbb{R}
ight) לכל s\in V סימון: יהי
L_{u,v} = \left\{egin{array}{ll} 0 & u=v \\ \ell((u,v)) & (u 
eq v) \land ((u,v) \in E) \\ \infty & (u 
eq v) \land ((u,v) 
eq E) \end{array} 
ight. מתקיים u,v \in V מתקיים באשר לכל L \in M_{|V|}\left(\mathbb{R}\right) אזי L \in M_{|V|}\left(\mathbb{R}\right) אזי באשר לכל יהי
```

 $.\delta_k\left(s
ight)=\delta_{k-1}\left(s
ight)*L$  מסקנה: יהי  $s\in V$  מטריצת מטריצת מטריצת ותהא ותהא  $s\in V$  מתקיים מסקנה:  $.D_{u,v}^{(k)}=\delta_k\left(u,v
ight)$  מתקיים  $u,v\in V$  באשר לכל באשר  $D^{(k)}\in M_{|V|}\left(\mathbb{R}
ight)$  אזי אי

```
D^{(k)}=L^k אזי מטריצת מטריצת L\in M_{|V|}\left(\mathbb{R}
ight) טענה: תהא
                                                             D^{(k)}=D^{(m)} אזי k,m\geq |V|-1 וחסר מעגלים שליליים ויהיו וממושקל \ell אזי מכוון וממושקל יהי
                                  D_{n,v}^{(|V|)} < 0 אזי שלילי שלילי במעגל שלילי ויהי עv \in V המופיע במעגל שלילי אזי וממושקל ענה: יהי
                                                                  .APSP מסקנה: תהא L \in M_{|V|}\left(\mathbb{R}
ight) מטריצת המשקל אזי וL \in M_{|V|}\left(\mathbb{R}
ight)
                                                             אלגוריתם חזקה איטרטיבית: תהא A\in M_n\left(\mathbb{R}
ight) אלגוריתם חזקה איטרטיבית אוי
function RepeatedSquaring(A, \star):
     (a_k \dots a_0) \leftarrow (n)_2
          B = B \star A
                                            .APSP פתרון לבעיית RepeatedSquaring (L,*) אזי מטריצת מטריצת מטריצת מטריצת מטריצת מטריצת המשקל מינה:
      \mathcal{O}\left(\left|V
ight|^3\log\left(\left|V
ight|
ight)
ight) הינה RepeatedSquaring (L,*) שענה: תהא L\in M_{\left|V\right|}\left(\mathbb{R}
ight) מטריצת המשקל אזי סיבוכיות זמן הריצה של
                                              מתקיים u,v\in V באשר לכל באשר F^{(k)}\in M_{n}\left(\mathbb{R}
ight) אזי און ויהי ויהי גרף מכוון ויהי גרף מכוון ויהי
                                            F_{u,v}^{(k)} = \min \left\{ \ell\left(\sigma\right) \mid \left(\sigma \in \left\{u 
ightarrow v
ight\}
ight) \wedge \left(עוברת דרך הצמתים עוברת למעט בהתחלה ובסוף \left[k\right]
F_{u,v}^{(0)}=L מתקיים u,v\in V באשר לכל באשר F^{(0)}\in M_n\left(\mathbb{R}
ight) מטריצת מטריצת מטריצת מטריצת מטריצת מטריצת מטריצת מטריצת המשקל אזי
                                          F_{u,v}^{(k)}=\min\left\{F_{u,v}^{(k-1)},F_{u,k}^{(k-1)}+F_{k,v}^{(k-1)}
ight\} אזי u,v\in[n] גרף מכוון ויהיו ([n] , E) איזי איזי
                                                 אלגוריתם פלויד־וורשאל: יהי ([n]\,,E) גרף מכוון ותהא אלגוריתם פלויד־וורשאל: יהי
function FloydWarshall (n, L):
              if (u \neq v) \wedge (L_{u,v} < \infty) then
                | \Pi_{u,v} \leftarrow u
                \Pi_{u,v} \leftarrow \text{None}
              for v \in [n] do
                   \begin{array}{c|c} \text{if } F_{u,k} + F_{k,v} < F_{u,v} \text{ then} \\ F_{u,v} \leftarrow F_{u,k} + F_{k,v} \\ \Pi_{u,v} \leftarrow \Pi_{k,v} \end{array}
```

 $B \leftarrow M_n(\mathbb{R})$ for  $i \in [k]$  do

end return B

if  $a_i = 1$  then

 $A = A \star A$ 

 $\Pi \leftarrow M_n([n])$ for  $u \in [n]$  do for  $v \in [n]$  do

else

end

end

return  $(F,\Pi)$ 

end

end end  $F \leftarrow L$ for  $k \in [n]$  do for  $u \in [n]$  do  $D^{(k)} = D^{(k-1)} * L$  אזי המשקל מטריצת מטריצת  $L \in M_{|V|}(\mathbb{R})$  מסקנה: תהא

```
.APSP פתרון לבעיית FloydWarshall (n,L) איי מטריצת המשקל אזי L\in M_n\left(\mathbb{R}
ight) פתרון לבעיית L\in M_n\left(\mathbb{R}
ight) מטריצת היי
\mathcal{O}\left(n^3
ight) הינה FloydWarshall (n,L) מטריצת המשקל אזי סיבוכיות זמן הריצה של בL\in M_n\left(\mathbb{R}
ight) הינה הינה L\in M_n\left(\mathbb{R}
ight)
                                                                    (u,v) 
otin E מתקיים u,v \in I עבורה לכל עבורה אזי I \subseteq V ארף אזי יהי G מתקיים
                            \min\left(i
ight)=\max\left\{w\left(I
ight)\mid\left(I\subseteq\left[i
ight]
ight)\wedge\left( בלתי תלויה w:\left[n
ight]
ightarrow\mathbb{R}_{>0} אזי w:\left[n
ight]
ightarrow\mathbb{R}_{>0} גרף שרוך ויהי
                                                        וכן \min (1) = w \, (1) וכן \min (0) = 0 אזי w : [n] \to \mathbb{R}_{\geq 0} וכן יהי גרף שרוך ויהי ויהי
                                                                                                                       mis(i) = max\{w(i) + mis(i-2), mis(i-1)\}\
                                                  \mathcal{O}\left(n
ight) אזי \min\left(n
ight) אזי w:[n]	o\mathbb{R}_{\geq0} אוירי שרוך אמן היצר על היבוכיות מסקנה: יהי
A_{f(i)}=B_i עבורה ממש וחח"ע המקיימת f:[|B|]	o [|A|] עבורה קיימת אזי B\in \Sigma^* אזי אלפבית ותהא אלפבית ותהא
                                                                                                                                                                              i \in [|B|] לכל
                                                                                  B \lhd A ותהא B \in \Sigma^* ותהא ותהא A \in \Sigma^* אלפבית תהא אזי
\max\{|C|\mid (C\in\Sigma^*)\land (C\lhd A)\land (C\lhd B)\} איי A,B\in\Sigma^* אלפבית ותהיינה \Sigma אלפבית ארוכה ביותר (LCS): יהי
    \text{lcs } (k,\ell) = \max \left\{ |C| \mid (C \lhd (A_1,\ldots,A_k)) \land (C \lhd (B_1,\ldots,B_\ell)) \right\} \text{ איז } \ell \leq |B| \text{ איז } k \leq |A| \text{ תהא } A,B \in \Sigma^* \text{ סימון: תהיינה } A,B \in \Sigma^* \text{ outs.}   \text{lcs } (k,\ell) = \begin{cases} 0 & (k=0) \lor (\ell=0) \\ \log(k-1,\ell-1)+1 & (k,\ell>0) \land (A_k=B_\ell) \end{cases} \text{ with } A,B \in \Sigma^* \text{ outs.}   \text{outs.} \text{ max} \left\{ \log(k-1,\ell-1)+1 & (k,\ell>0) \land (A_k=B_\ell) \\ \max\{\log(k-1,\ell),\log(k,\ell-1)\} & (k,\ell>0) \land (A_k\neq B_\ell) \end{cases}   \text{lcs } \left( |A| \cdot |B| \right) \text{ in the constant of } A,B \in \Sigma^* \text{ outs.}   \text{adjets.} \text{ adjets.} \text{ and } C \in \mathbb{R}^* \text{ outs.} 
\max\left\{|C|\mid (C\lhd A)\land (orall i.C_{i-1}\prec C_i)
ight\} אזי אוי A\in \Sigma^* אאי אלפבית בעל סדר היינ\Sigma אלפבית היי\Sigma אלפבית בעל סדר אותהא
                                                                                A, \operatorname{sort}(A) של LCS של בעיית של LIS טענה: תהא A \in \Sigma^* אזי בעיית
                                  .lenlis (k)=\max\left\{|X|\mid ((A_1,\ldots,A_k)\, של של ווא X)\wedge (A_k מסתיים עם X)
ight\} אזי אזי A\in\Sigma^* אווי תהא
                                                         .lenlis (k)=\max_{i\in[k-1]}\left\{ \mathrm{lenlis}\left(i\right)\mid A_{i}\prec A_{k}
ight\} וכך וכוווs (1)=1 אזי A\in\Sigma^{*} איזי A\in\Sigma^{*}
                                                            \pilis (k)=rg\max{\{	ext{lenlis}\,(i)\mid A_i\prec A_k\}} וכן \pilis (1)=	ext{None} אזי A\in\Sigma^* סימון: תהא
LIS מסקנה: תהא A \in \Sigma^* ויהי (x_{\pi \mathrm{lis}(\ell)(k)}, \dots, x_{\pi \mathrm{lis}(2)(k)}, x_{\pi \mathrm{lis}(k)}, x_k) איי אי k = rg \max \{ \mathrm{lenlis}(1), \dots, \mathrm{lenlis}(|A|) \} פתרון של
                                                                                                                                              \mathcal{O}\left(\left|A
ight|^{2}
ight) בעל סיבוכיות זמן ריצה
                                                                                            .min lis (m)=\min\left\{x_k\mid 	ext{lenlis}\left(k
ight)=m
ight\} אזי A\in\Sigma^* אינון: תהא
                                                                                                                                 . עולה ממש \min lis אזי A \in \Sigma^* עולה ממש
                      \mathcal{O}\left(|A|\cdot\log\left(|A|
ight)
ight) אזי אמן ריצה (\min\operatorname{lis}\left(1
ight),\ldots,\min\operatorname{lis}\left(\ell
ight) אזי אזי (\min\operatorname{lis}\left(1
ight),\ldots,\min\operatorname{lis}\left(\ell
ight) אזי אזי (\min\operatorname{lis}\left(1
ight)
                      \operatorname{costp}(T) = \sum_{i=1}^n \left( p_i \cdot \operatorname{depth}_T(x_i) \right) אזי \{x_1 \dots x_n\} איי עץ חיפוש בינארי עץ T ויהי ויהי p_1 \dots p_n \in (0,1] איי
                         מינימלי. Costp\left(T\right) עבורו בינארי אופטימלי: יהיו p_1\dots p_n\in(0,1] אזי עץ חיפוש בינארי סטטי אופטימלי: יהיו
                   \operatorname{costp}(T) = (\sum_{i=1}^n p_i) + \operatorname{costp}(T.\operatorname{left}) + \operatorname{costp}(T.\operatorname{right}) אינה: יהיו עץ חיפוש בינארי אזי p_1 \dots p_n \in (0,1] ויהי
מסקנה: יהיו T.left, T.right מסקנה: יהיו בינארי סטטי אופטימלי פתרון לבעיית עץ ויהי T פתרון לבעיית עץ ויהי
                                                                                                                                                     חיפוש בינארי סטטי אופטימלי.
                                                                                                              .pp (i,j)=\sum_{k=i}^{j}p_{k} אזי p_{1}\dots p_{n}\in(0,1] סימון: יהיו
                  \operatorname{cp}(i,j) = \min \left\{ \operatorname{costp}(T) \mid \{x_i \dots x_j\} \; סימון: יהיו p_1 \dots p_n \in (0,1] ויהיו אזי x_1 \dots x_n אזי ויהיו
                                                              וכן \operatorname{cp}\left(i,i\right)=p_{i} וכן \operatorname{cp}\left(i,i-1\right)=0 אזי x_{1}\ldots x_{n} ויהיו p_{1}\ldots p_{n}\in\left(0,1\right] וכן
                                                                                                  .cp(i, j) = pp(i, j) + \min_{i \le k \le j} (cp(i, k - 1) + cp(k + 1, j))
                                          מסקנה אלגוריתם לבעיית עץ חיפוש בינארי סטטי אופטימלי: יהיו p_1 \ldots p_n \in (0,1] ויהיו x_1 \ldots x_n אזי
```

```
function OSBST(pp):
     K, C \leftarrow \text{List}([n]^2)
     for i \leftarrow [n+1] do
      C(i, i-1) \leftarrow 0
     end
     for d \leftarrow \{0, \ldots, n-1\} do
         for i \leftarrow [n-d] do
              C(i, i+d) \leftarrow \infty
               for k \leftarrow \{i, \dots, i+d\} do
                    t \leftarrow \operatorname{pp}(i,j) + C(i,k-1) + C(k+1,j)
                    if t < C(i, j) then
                         C(i,j) \leftarrow t
                        K(i,j) \leftarrow k
              end
          end
     end
```

מסקנה: יהיו  $p_n\in(0,1]$  אזי  $p_n$  (pp) משרה פתרון לבעיית עץ חיפוש בינארי סטטי אופטימלי.  $p_1\dots p_n\in(0,1]$  מסקנה: יהיו  $p_1\dots p_n\in(0,1]$  אזי (OSBST (pp) בעל סיבוכיות זמן ריצה ( $p_1\dots p_n\in(0,1]$  אזי  $p_1\dots p_n\in(0,1]$  הערה: קיים אלגוריתם קנות' לבעיית עץ חיפוש בינארי סטטי אופטימלי בסיבוכיות זמן ריצה ( $p_1\dots p_n\in(0,1]$  הערה:  $p_1\dots p_n\in(0,1]$  מקסימלית וכן  $p_1\dots p_n\in(0,1]$  מקסימלית וכן  $p_2\dots p_n\in(0,1]$  מקסימלית וכן  $p_1\dots p_n\in(0,1]$  מקסימלית וכן  $p_2\dots p_n\in(0,1]$  מקסימלית וכן  $p_2\dots p_n\in(0,1]$  מקסימלית וכן  $p_2\dots p_n\in(0,1]$  אזי  $p_2\dots p_n\in(0,1]$  מקסימלית וכן  $p_2\dots p_n\in(0,1]$ 

אזי  $v_1 \dots v_n \geq 0$  ויהיו  $W, w_1 \dots w_n > 0$  יהיו הגב: יהיו שבר תרמיל לבעיית שבר אלגוריתם אלגוריתם

```
function Fractional Knapsack (W, w_1, \dots, w_n, v_1, \dots, v_n):
     f \leftarrow ([n] \rightarrow [0,1])
     P \leftarrow \text{List}([n] \times \mathbb{R})
     for i \leftarrow [n] do
           P(i) \leftarrow (i, \frac{v_i}{w_i})
          f(i) \leftarrow 0
     end
     P \leftarrow \operatorname{sort}(P) // Sort from high to low based on second coordinate.
     t \leftarrow 0
     i \leftarrow 1
     while (t < W) \land (i \le n) do
           j \leftarrow P(i)[0]
           if t + w_j \leq W then
                f(j) \leftarrow 1
               t \leftarrow t + w_i
                f(j) \leftarrow \frac{W-t}{m}
     end
     return f
```

```
.bknap (k,W)=\max\left\{\sum_{i\in S}v_i\mid (S\subseteq[k])\wedge\left(\sum_{i\in S}w_i\leq W\right)\right\} אזי W,w_1\dots w_n,v_1\dots v_n\geq 0 טענה: יהיו w_1\dots w_n,v_1\dots v_n\geq 0 אזי w_1\dots w_n,v_1\dots v_n\geq 0 .bknap (0,m)=0 אזי m\geq 0 .bknap (i,0)=0 אזי i\in [n] .bknap (i,0)=0 אזי i\in [n] .bknap (i,m)=\left\{\max_{\substack{\text{bknap}(i-1,m)\\\text{wis} m}}\frac{w_i>m}{w_i\leq m} אזי i\in [n] אזי m\geq 0 .correction adaptation of w_i אזי חישוב w_i>m אזי w_i>m אזי w_i>m אזי חישוב w_i>m אזי w_i>m אזי w_i=m אזי w_i>m אזי חישוב w_i>m אזי w_i=m אונוריים אונוניים אונוניים
```

```
function ZeroOneKnapsack(W, w_1, \ldots, w_n, v_1, \ldots, v_n):
    w \leftarrow W
    S \leftarrow \operatorname{Set}([n])
    S \leftarrow \varnothing
    while (k>0) \wedge (w>0) do
         if bknap(k, w) \neq bknap(k-1, w) then
             S \leftarrow S \cup \{k\}
             k \leftarrow k - 1
             w \leftarrow w - w_k
          k \leftarrow k-1
     end
          פתרון לבעיית 0/1 פתרון לבעיית פרסOneKnapsack (W,w_1\dots w_n,v_1\dots v_n) אזי אי W,w_1\dots w_n,v_1\dots v_n\geq 0 מסקנה: יהיו
                                                        (V,E,c,s,t) אזי s,t\in V ותהיינה c>0 וממושקל מכוון וממושקל
                                                                                        c אזי זרימה אזי (V,E,c,s,t) אוי תהא פונקציית קיבולת:
                                                                                            s אזי ארימה ארי (V,E,c,s,t) אוי מקור: תהא
                                                                                               t אזי ארימה ארי רשת (V,E,c,s,t) אוי הוד קודקוד בור:
                                           עודף \chi_f:V	o\mathbb{R} אזי f:E	o\mathbb{R}_{\geq 0} רשת זרימה ותהא רשת (V,E,c,s,t) אזי עודף זרימה: תהא
                                                                 .\chi_f\left(v
ight)=\sum_{\substack{u\in V\\ (u,v)\in E}}f\left((u,v)
ight)-\sum_{\substack{u\in V\\ (v,u)\in E}}f\left((v,u)
ight) עבורה f:E	o\mathbb{R}_{\geq 0} רשת זרימה אזי f:E	o\mathbb{R}_{\geq 0}
                                                                                                                          f \leq c חסם קיבולת: •
                                                                                     \chi_f(v)=0 מתקיים v\in V\setminus\{s,t\} שימור זרם: לכל
                          . מקסימלית: תהא \chi_f(t) מקסימלית: רשת איי פונקציית הירימה איי פונקציית הארימה המקסימלית: תהא (V,E,c,s,t) מקסימלית:
                         s \in S וכן S \uplus T = V וכן S,T \subseteq V באשר אזי (S,T) רשת זרימה אזי (V,E,c,s,t) וכן ישר יאר מתד: s-t
                        E\left(S,T
ight)=\left\{ (u,v)\in E\mid (u\in S)\wedge (v\in T)
ight\} אזי s-t חתך ויהי ארימה ויהי רשת ארימה G רשת חוצות: תהא
                     E\left(T,S
ight)=\left\{ \left(u,v
ight)\in E\mid\left(u\in T
ight)\wedge\left(v\in S
ight)
ight\} אזי \left(S,T
ight) חתך הימה הוריות: תהא G רשת זרימה ויהי
                                                                      .c\left(S,T
ight)=\sum_{e\in E\left(S,T
ight)}c\left(e
ight) אזי s-t חתך ואי הי הי יהי
                                         f\left(S,T
ight)=\sum_{e\in E\left(S,T
ight)}f\left(e
ight)-\sum_{e\in E\left(T,S
ight)}f\left(e
ight) אזי s-t חתך אזי (S,T) אזי מני חתך: יהי
                                                                               |f|=f\left(V\setminus\{t\},\{t\}\right) אוימה: תהא f זרימה: תהא אזי
                                                                                  |f|=f\left(S,T\right) אזי s-t חתך אזי ויהי f ארימה ויהי זרימה ויהי
                                                                                             |f|=f\left(\left\{ s\right\} ,V\backslash\left\{ s\right\} \right) איי זרימה איי f ארימה: תהא
                                                                            f\left(S,T
ight)\leq c\left(S,T
ight) אזי s-t חתך אויהי ויהי זרימה ויהי f ארימה איי
                                                                 f\left(S,T
ight)=c\left(S,T
ight) עבורו f\left(S,T
ight) אזי אויה אזי f\left(S,T
ight) אזי איימה ויהי
                                                                                                                            . זרימה מקסימלית f
                                                                                 c\left((S,T)\right) \leq c\left((A,B)\right) מתקיים (A,B) s-t לכל חתך
                                              e \in P לכל f\left(e
ight) < c\left(e
ight) באשר באשר אזי לכל זרימה אזי f לכל מסלול ניתן להגדלה :s-t מסלול ניתן להגדלה
g_{\restriction_{E \setminus P}} = f_{\restriction_{E \setminus P}} עבורה g עבורה אזי פיימת פונקציית איי איי פיימת מסלול: תהא ארימה ויהי ויהי ארימה ויהי ויהי P \in \{s \to t\} מסלול ניתן להגדלת מסלול:
                                                                                                                                           |f| < |g| וכן
                                                                   .s-t מסלול ניתן להגדלה עבורה לא קיים מסלול ניתן להגדלה f ארימה פונקציית זרימה מסלול ניתן
                                                               e^{-1} אזי e^{-1} \in E אבורה עבורה פכוון ותהא אנטי־מקבילה: יהי
     באשר (V, E_f, c_f, s, t) באיר זרימה היור אנטי־מקבילות ותהא f זרימה אזי רע, רשת זרימה אורית: תהא
                                                                                                   .E_f = \{e \in E \mid c(e) > f(e)\} \cup E^{-1} \bullet
```

```
c_f\left(e
ight)=c\left(e
ight)-f\left(e
ight)+f\left(e^{-1}
ight) אזי e\in E אהיבולת: תהא
```

 $.E_f = \{e \in E \mid c_f(e) > 0\} \bullet$ 

 $.c\left((u,v)
ight)=0$  אזי (u,v)
otin E עבורם  $u,v\in V$  אזי הערה: יהיו

. הינה רשת הזרימה השיורית.  $G_f$  הינה רשת זרימה ותהא f ארימה השיורית. תהא

 $G_f$  מסלול ניתן לשיפור  $P \in \{s o t\}$  תהא  $P \in \{s o t\}$  בגרף זרימה אזי מסלול ניתן לשיפור בהא רשת זרימה ותהא

 $.c_{f}\left(P
ight)=\min\left\{ c_{f}\left(e
ight)\mid e\in P
ight\}$  אזי s-t מסלול ניתן מסלול מסלול: תהא א זרימה זיימה ויהי P ארימה זיימה אזי מסלול: מחסום

 $e\in E\left(G
ight)$  לכל  $f_{P}\left(e
ight)=\left\{egin{array}{ll} f(e)+c_{f}(P) & e\in P \\ f(e)-c_{f}(P) & e^{-1}\in P \end{array}
ight.$  איי  $f_{P}\left(e
ight)=\left\{egin{array}{ll} f(e)-c_{f}(P) & e^{-1}\in P \\ f(e) & else \end{array}
ight.$  איי  $f_{P}\left(e
ight)=\left|f\right|+c_{f}\left(e
ight)$  וכן  $f_{P}\left(e
ight)$  איי  $f_{P}\left(e
ight)$  ארימה של  $f_{P}\left(e
ight)$  וכן  $f_{P}\left(e
ight)$  איי  $f_{P}\left(e
ight)$  ארימה של  $f_{P}\left(e
ight)$  וכן  $f_{P}\left(e
ight)$ 

משפט: תהא f זרימה התב"ש

- Gזרימה מקסימלית ב'  $f \bullet$
- .s-t מסלול ניתן מסלול מינו מחלול  $P \in \{s \to t\}$  מתקיים כי  $P \in \{s \to t\}$ 
  - G- מינימלי ל-s-t חתך (S,T) מינימלי ל-

 $\max\{|f|\mid$  זרימה  $f\}=\min\{c(S,T)\mid$  חתך  $(S,T)\}$  חתר ורימה איי ורימה G ארימה: תהא איי ורימה מקסימלית קיבולת מינימלית: תהא אלגוריתם פורד־פלקרסון: תהא (V,E,c,s,t) רשת זרימה אזי

```
function FordFulkerson(V, E, c, s, t):
```

```
f \leftarrow (E \to \mathbb{R})
f \leftarrow 0
while True do
       G_f \leftarrow \text{ResidualNetwork}(G, c, s, t, f) // Construct it like any graph.
       \begin{array}{l} \pi_{G_f} \leftarrow \operatorname{BFS}(G,s) \\ \text{if } \{s \to t\} \cap \pi_{G_f} = \varnothing \text{ then } \operatorname{return } f \end{array}
            P \leftarrow \{s \rightarrow t\} \cap \pi_{G_f} // The path is taken from \pi_{G_f} . f \leftarrow f_P
end
```

.FF = FordFulkerson (V, E, c, s, t) אזי רשת אוי (V, E, c, s, t) רשת אוי (V, E, c, s, t)

 $f\left(E
ight)\subset\mathbb{N}$  באשר  $f\left(E
ight)\subset\mathbb{N}$  אזי קיימת ארימה מקסימלית  $\left(V,E,c,s,t
ight)$  באשר אזי פונימת הרימה באשר

מתקיים FF מתקיים אזי בכל איטרציה אוי בענה: רשת ארימה באשר (V,E,c,s,t) אזי בכל

- G זרימה של  $f \bullet$ 
  - $f(E) \subseteq \mathbb{N} \bullet$
  - $.c_{f}(P) > 1 \bullet$

משפט: תהא  $f(E)\subseteq\mathbb{N}$  רשת ארימה באשר  $f(E)\subseteq\mathbb{N}$  ותהא אוי רימה באשר (V,E,c,s,t) אאי

- FF פתרון לבעיית הזרימה המקסימלית לכל בחירת מסלולים באלגוריתם.
  - עושה לכל היותר |f| שיפורי מסלול. FF ullet
    - .FF  $(E) \subseteq \mathbb{N} \bullet$

מסקנה: תהא  $f\left(E
ight)\subseteq\mathbb{N}$  אזי סיבוכיות זמן הריצה  $c\left(E
ight)\subseteq\mathbb{N}$  אזי סיבוכיות זמן הריצה (V,E,c,s,t) איז חיבה מסקנה: תהא  $\mathcal{O}\left(\left|E\right|\left|f\right|\right)$  של FF של

 $|e_1\cap e_2| 
eq 1$  מתקיים  $e_1,e_2\in M$  עבורה לכל  $M\subseteq E\left(G
ight)$  אזי מכוון אזי מכוון אזי  $M\subseteq E\left(G
ight)$ 

 $\operatorname{arg\,max}\left\{|M|\mid G$  בעיית זיווג מקסימלי: יהי G גרף לא מכוון אזי M זיווג של

 $M=V\left( G
ight)$  עבורו  $M\subseteq E\left( G
ight)$  איווג מושלם: יהי G גרף לא מכוון אזי זיווג

 $|e\cap A|=|e\cap B|=1$  מתקיים  $e\in E\left(G
ight)$  מים וויהיו  $A\oplus B=V\left(G
ight)$  עבורם  $A,B\subseteq V\left(G
ight)$  מתקיים A מתקיים מים A $G_R = B$  וכן  $G_L = A$  אזי

 $V^{\perp}=V\left(G
ight)\cup\left\{ s,t
ight\}$  אזי  $s,t
otin V\left(G
ight)$  ויהיו איז לא מכוון ויהיו מימון: יהי G גרף דו־צדדי לא מכוון ויהיו

 $E^{
ightarrow}=\{\langle v,u
angle \mid (\{v,u\}\in E(G)) \land (v\in G_L) \land (u\in G_R)\}$  סימון: יהי G גרף דו־צדדי לא מכוון אזי

 $E^{\perp}=(\{s\} imes G_L)\cup E^{
ightarrow}\cup (G_R imes \{t\})$  אזי  $s,t\notin V\left(G
ight)$  ויהיו לא מכוון ויהיו

```
.c_{\lceil_{E}
ightarrow}^{\perp}=\infty וכן .c_{\lceil_{\left\{s
ight\}	imes G_{L}
ight)\cup\left(G_{R}	imes\left\{t
ight\}
ight)}^{\perp}=1 המוגדרת .c^{\perp}:E^{\perp}
ightarrow\mathbb{R}_{+} אזי .t\notin V\left(G
ight) וכן .t\in\mathcal{C}_{\lceil_{\left\{s
ight\}	imes G_{L}
ight)\cup\left(G_{R}	imes\left\{t
ight\}
ight)}^{\perp}=1 המוגדרת .t\in\mathcal{C}_{\lceil_{\left\{s
ight\}	imes G_{L}
ight)\cup\left(G_{R}	imes\left\{t
ight\}
ight)}^{\perp}=1
                                                                                                                  .G^\perp = (V^\perp, E^\perp, c^\perp, s, t) אזי א s,t \notin V\left(G
ight) ויהיו לא מכוון ויהיו מכוון יהי לא מרף דו־צדדי איזי מכוון ויהיו
                                                                                                                                      אלגוריתם לבעיית זיווג מקסימלי בגרף דו־צדדי: יהי G גרף דו־צדדי לא מכוון אזי
function BMMF(G):
           \begin{array}{l} (s,t) \not\leftarrow V\left(G\right) \\ G^\perp \leftarrow (V^\perp, E^\perp, c^\perp, s, t) \end{array} 
           f \leftarrow \text{FordFulkerson}(G^{\perp})
          return \{e \in E(G) \mid f(e) = 1\}
                                                                                                                                                              . אינו זיווג מקסימלי. אווג מקסימלי. אווג מקסימלי אווג מקסימלי. ארף דו־צדדי לא מכוון אזי G
                                                                                                          \mathcal{O}\left(|E|\cdot|V|
ight) אמן ריצה אמן בעל סיבוכיות אזי מכוון אזי אזי לא מכוון אזי מענה: יהי G גרף דו־צדדי לא מכוון אזי
                 \max\left\{|M|\mid G איווג של M
ight\}=\max\left\{|f|\mid G^\perp ארימה של s,t
otin V(G) איווג של איווג של s,t
otin V(G) איווג של
                                                                                                       e\cap C
eq \emptyset מתקיים e\in E מתקיים עבורה לכל עבורה אזי מניטון אזי מכוון אזי C\subset V\left( G
ight)
                                                                                                   rg \min \left\{ |C| \mid G 
ight. בעיית כיסוי צמתים של G גרף לא מכוון אזי C כיסוי צמתים של מינימלי: יהי
                                                            \max\{|M|\mid G איווג של M\}\leq \min\{|C|\mid G למה: יהי Mגרף דו־צדדי לא מכוון אזיC כיסוי צמתים של
                                                                                                                   אלגוריתם לבעיית כיסוי צמתים מינימלי בגרף דו־צדדי: יהי G גרף דו־צדדי לא מכווו אזי
function BMVC(G):
          (M, s, t, G^{\perp}, f) \leftarrow \text{BMMF}(G)
           C \leftarrow V(G)
           for \{u,v\} \in M \cap (G_L \times G_R) do
                    \inf\left\{\tau:s\to v \ \middle| \ G_f^\perp \ \text{ מסלול בגורף} \ \tau\right\} \neq \varnothing \ \text{then} \\ \mid \ C\leftarrow C\cup \{v\}
                     else
                     \mid C \leftarrow C \cup \{u\}
           end
           return C
                                                                                                                                                                  . אינו כיסוי צמתים BMVC (G) אזי לא מכוון אזי דו־צדדי ארף דו־צדדי אינו מענה: יהי
                                                                                                                             .|\mathrm{BMVC}\,(G)|=|M| אווג מקסימלי אוי זיווג מכוון ויהי א גרף דו־צדדי איז גרף איווג מכוון ויהי M
                                                                                                                                        . מסקנה: יהי G גרף דו־צדדי לא מכוון אזי BMVC (G) אזי לא מינימלי גרף דו־צדדי מינימלי
                                                        \max\{|M|\mid G איווג של M\}=\min\{|C|\mid G משפט: יהי Mגרף דו־צדדי לא מכוון אזיC כיסוי צמתים של
                                                   \mathrm{DP}_{s,t} = \max \{n \in \mathbb{N} \mid t^{-t} גרף מכוון ויהיו s,t \in V אזי לקיימים n מסלולים זרים בקשתות מ־s,t \in V אזי אזי לקיימים
                     \mathtt{DE}_{s,t} = \min \left\{ n \in \mathbb{N} \mid tל מ־ס מסלול מ' קיימות שאם נסירן אזי s,t \in V אזי איני s,t \in V סימון: יהי
                                                                                                                                                                                          (V,E,1,s,t) אזי s,t\in V רשת G יהי יהי G יהי
                                                                                                                          \mathrm{DP}_{s,t} = \max \left\{ |f| \mid \mathsf{O/1} \mid \mathsf{DP}_{s,t} = \mathsf{MEX} \left\{ |f| \mid \mathsf{O/1} \mid \mathsf{DP}_{s,t} \mid \mathsf{S}, t \in V \mid \mathsf{S}, t \in 
                                                                                         	ext{DE}_{s,t} = \min \left\{ c\left(S,T
ight) \mid 	ext{0/1} ברשת s-t חתך s-t אזי s,t \in V אזי אויהיו G גרף מכוון ויהיו
                                                                                                                                                               \mathtt{DP}_{s,t} = \mathtt{DE}_{s,t} אזי s,t \in V מסקנה משפט מנגר: יהי
                                                                          N\left(A
ight)=\{y\in G_R\mid (A	imes\{y\})\cap E
eqarnothing\} אזי A\subseteq G_L אזי לא מכוון ותהא G יהי G יהי G יהי
משפט החתונה/הול: יהי G גרף דו־צדדי לא מכוון באשר |G_L|=|G_R| אזי (קיים זיווג מושלם בGלכל G מתקיים A\subseteq G
                                                                                                                                                                                                                                                                                                                  |A| \leq |N(A)|
                                  uאזי גרף מכוון G אזי גרף מכוון u,v\in G קיימים u מסלולים זרים בקשתות מu אזי גרף u אזי גרף מכוון u
```

אלגוריתם לבדיקת k־קשירות בקשתות: יהי ויהי  $k\in\mathbb{N}_+$  גרף מכוון אזי

```
function kConnected(k, G):
    for u \in V \setminus \{v\} do
        /st The following FordFulkerson calls will return True if the flow size is bigger then k after k augmenting
            paths else False
        b_1 \leftarrow \text{FordFulkerson}(V, E, 1, v, u)
        b_2 \leftarrow \text{FordFulkerson}(V, E, 1, u, v)
        if (\neg b_1) \lor (\neg b_2) then return False
    return True
                                   .(kConnected (G)= True)\Longleftrightarrowטענה: יהי k\in\mathbb{N}_+ אויהי k\in\mathbb{N} גרף מכוון אזי k\in\mathbb{N}_+
                            עבורה f:E	o\mathbb{R}_{\geq 0} רשת זרימה אזי (V,E,c,s,t) עבורה ארימה זרימה/קדם ארימה פונקציית פרה־זרימה
                                                                                                                 f \leq c חסם קיבולת: •
                                                                                           \chi_f(v) \geq 0 מתקיים v \in V \setminus \{s, t\} •
x_f(v)>0 צומת גולשת/בעלת עודף זרימה: תהא v\in V\setminus \{s,t\} רשת זרימה ותהא פונקציית קדם זרימה אזיv\in V\setminus \{s,t\} עבורה
E_f = \{e \in E \mid c\left(e\right) > f\left(e\right)\} \cup E^{-1} ארימה אזי דימה חסרת קשתות אנטי־מקבילות ותהא f קדם זרימה אזי (V, E, c, s, t) רשת זרימה חסרת קשתות אנטי־מקבילות ותהא
E_f = E_f = \{e \in E \mid c_f(e) > 0\} רשת ארימה בעלת קשתות אנטי־מקבילות ותהא f קדם זרימה אזי (V, E, c, s, t) רשת זרימה בעלת קשתות אנטי־מקבילות ותהא
c_f\left(e
ight) =  המוגדרת הקיבולת: תהא c_f:E_f	o\mathbb{R}_+ אזי קדם זרימה ותהא f קדם ירימה אזי ועהא ועהא אוגדרת רעש. רשת ארימה ותהא

\cdot \begin{cases} c(e) - f(e) & e \in E \\ f(e^{-1}) & e \in E^{-1} \end{cases}

                \Delta_{u,v} = \min\left\{\chi_f\left(u\right), c_f\left((u,v)\right)\right\} אזי u,v \in V בימון: תהא לועם זרימה תהא לועם זרימה תהא לועם זרימה ויהיו
                                             אזי u,v\in V אזי ארימה f קדם ארימה (V,E,c,s,t) אאי אלגוריתם ארימה ויהיו
function Push ((V, E, c, s, t), f, u, v):
    f^* \leftarrow f
    if (u,v) \in E then
       f^*((u,v)) \leftarrow f((u,v)) + \Delta_{u,v}
    if (v, u) \in E then
     f^*((v,u)) \leftarrow f((v,u)) - \Delta_{u,v}
    return f^*
```

.Push (f,u,v)= Push ((V,E,c,s,t),f,u,v) אזי  $u,v\in V$  איי ארימה תהא f קדם ארימה תהא (V,E,c,s,t) רשת ארימה תהא פדם זרימה. Push (f,u,v) אזי  $u,v\in V$  איזי אויימה תהא f קדם ארימה תהא (V,E,c,s,t) איזי רימה.  $\chi_{\mathsf{Push}(f,u,v)}\left(u
ight)=\chi_{f}\left(u
ight)-\Delta_{u,v}$  אזי איזי  $u,v\in V$  טענה: תהא t קדם זרימה תהא t קדם זרימה ויהיו .Push (f,u,v) אזי  $\Delta_{u,v}=c_f\left((u,v)
ight)$  עבורם  $u,v\in V$  אזי קדם זרימה תהא t קדם ארימה  $\Delta_{u,v}=c_f\left((u,v)
ight)$  עבורם עבורה  $h:V o\mathbb{N}$  אזי  $h:V o\mathbb{N}$  רשת זרימה ותהא f קדם זרימה אזי (V,E,c,s,t) עבורה

- $.h(s) = |V| \bullet$ 
  - $.h(t) = 0 \bullet$
- $h(u) \leq h(v) + 1$  אזי  $(u,v) \in E_f$  יהי

```
function Relabel ((V, E, c, s, t), f, h, u):
    h^* \leftarrow h
    h^*(u) \leftarrow \min\{h(v) \mid (u, v) \in E_f\} + 1
    return h*
```

Relabel (f,h,u)=v אזי $u\in V$  אזי מונקציית גובה ותהא t קדם זרימה תהא t קדם זרימה תהא t פונקציית גובה ותהא .Push ((V, E, c, s, t), f, h, u)  $u\in V$  טענה: תהא h פונקציית גובה ותהא t רימה תהא  $t\in V$  רשת זרימה ותהא  $t\in V$  אזי

- .Relabel (f,h,u)  $(u) \leq \text{Relabel}$  (f,h,u) (v)+1 אזי  $(u,v) \in E_f$  יהי •
- .Relabel (f, h, u)  $(w) \leq$  Relabel (f, h, u) (u) + 1 אזי  $(w, u) \in E_f$  יהי

Relabel (f,h,u) אזי  $u\in V\setminus\{s,t\}$  תהא גובה ותהא פונקציית גובה f קדם זרימה תהא t קדם זרימה אזי t פונקציית גובה.

 $(u,v)\in E_f$  אזי קיימת  $u\in V\setminus \{s,t\}$  אוי פונקציית גובה ותהא פונקציית אוי קיימת v רשת זרימה תהא למה: תהא פונקציית גובה ותהא (v,t,t,t,t) רשת זרימה הא קדם אוי קיימת הא פונקציית גובה ותהא (v,t,t,t,t,t) רשת זרימה הא פונקציית גובה ביחס ל־v,t,t,t,t,t

 $h\left(u
ight) \leq h\left(v
ight) + \delta_{G_f}\left(u,v
ight)$  אזי  $u,v \in V$  אזי גובה ותהיינה h פונקציית גובה h קדם ארימה תהא h קדם דרימה תהא h פונקציית גובה ויהי  $u \in V$  עבורו קיים מסלול מ־u למה: תהא h קדם ארימה תהא h פונקציית גובה ויהי  $u \in V$  עבורו קיים מסלול מ־u למה:  $h\left(u
ight) \leq |V| - 1$  אזי  $u \in V$ 

 $G_f$ ב ב־u מסקנה: תהא  $u \in V$  עבורו קיים מסלול מ־u ל־u ל־u ל־u ל־u ל־u ל־u ל־u ל־u ל-u ל-u עבורו איי ל $u \in V$  עבורו איז ל $u \in V$  ל-u ל-u

 $G_f$ ב ל־ל ב־ל מסקנה: תהא (V,E,c,s,t) רשת ארימה תהא f קדם ארימה ותהא f פונקציית גובה אזי לא קיים מסלול מ־f למה: תהא f רשת ארימה תהא f קדם ארימה ותהא f באשר f אזי קיים מסלול f מ־f למה: תהא f לכל f פרf לכל f ב־f לכל f

```
function GoldbergTarjan((V, E, c, s, t)):
```

```
\begin{array}{l} f \leftarrow (E \rightarrow \mathbb{R}_+) \\ f \leftarrow 0 \\ \text{for } (s,v) \in E \text{ do} \\ \mid f((s,v)) \leftarrow c((s,v)) \\ \text{end} \\ h \leftarrow (V \rightarrow \mathbb{N}) \\ h \leftarrow 0 \\ h(s) \leftarrow |V| \\ \text{while } \{u \in V \backslash \{s,t\} \mid \chi_f(u) > 0\} \neq \varnothing \text{ do} \\ \mid u \leftarrow \{u \in V \backslash \{s,t\} \mid \chi_f(u) > 0\} \\ \mid \text{if } \{(u,v) \in E_f \mid h\left(u\right) = h\left(v\right) + 1\} \neq \varnothing \text{ then} \\ \mid (u,v) \leftarrow \{(u,v) \in E_f \mid h\left(u\right) = h\left(v\right) + 1\} \\ \mid f \leftarrow \text{Push}(f,u,v) \\ \mid \text{else} \\ \mid h \leftarrow \text{Relabel}(f,h,u) \\ \mid \text{end} \\ \text{return } f \end{array}
```

 $f_s\left((u,v)
ight)=\left\{egin{array}{l} c((u,v))&u=s\\0&else\end{array}
ight.$  המוגדרת  $f_s:E o\mathbb{R}_+$  רשת זרימה אזי (V,E,c,s,t) המוגדרת  $\mathbb{1}_s:U=\{0,E,c,s,t\}$  רשת זרימה אזי  $\mathbb{1}_s:V o\mathbb{N}$  המוגדרת רשת זרימה אזי (V,E,c,s,t) רשת זרימה אזי  $|V|\cdot\mathbb{1}_s$  פונקציית גובה וכן לא קיים מסלול מ־(V,E,c,s,t) למה: תהא (V,E,c,s,t) רשת זרימה אזי לאחר כל איטרציה של GoldbergTarjan מתקיים

- . הינה קדם זרימה f ullet
  - .פונקציית גובה  $h \bullet$
- $.G_f$ ב ל־ל מ־s לא קיים מסלול מ-s

. פעמים איז Relabel קוראת לפונקציה GoldbergTarjan איז ארימה איז רשת לכל תהא לפונקציה על אוי רשת לימה: תהא למה: עלמה: עלמה איז רשת לימה אוי

. פעמים  $2\left|E\right|\left|V\right|$  ביותר לכל היותר מבצעת מבצעת אזי GoldbergTarjan בעמים רשת ארימה אוי רשת למה:

. פעמים פעמים אזי ארימה אזי הרימה אזי GoldbergTarjan מבצעת דחיפה לא מרווה לכל היותר (V,E,c,s,t) רשת ארימה אזי GoldbergTarjan הינה ארימה מקטימלית. (V,E,c,s,t) רשת ארימה אזי העובט הינה ארימה מקטימלית.

 $\mathcal{O}\left(\left|E\right|\cdot\left|V\right|^2\right)$  רשת זרימה אזי ניתן לממש את GoldbergTarjan עם ניתן ריצה (V,E,c,s,t) רשת זרימה אזי ניתן לממש את עם GoldbergTarjan בסיבוכיות זמן ריצה (V,E,c,s,t) רשת זרימה אזי ניתן לממש את V,E,c,s,t עם V,E,c,s,t הערה: תהא (V,E,c,s,t) ריצה  $\mathcal{O}\left(\left|E\right|\left|V\right|\log\left(\left|V\right|\right)\right)$