GRAFOVÉ ALGORITMY

2004/2005 - 1. termín

1. Tranzitivní obal.

Konečná uspořádaná množina $\mathcal{P} = (P, \leq)$ je zadána pomocí relace pokrývání $\prec$. Výpočet relace $\leq$ algoritmem Floyd – Warshall vezme čas ....

Simonův algoritmus.

Nechť $n = |P|,\ c = |\prec|,\ o = |\leq|$. Nechť množina P je reprezentována seznamem odpovídajícímu nějakému topologickému uspořádání grafu $(P, \prec)$. Nechť $ord\,[x] = i$ značí, že $x \in P$ je zde i-tým prvkem. Pro každé $x \in P$ máme vypočítat seznam $F[x]$ reprezentující množinu $f(x) = \{\, y \in P \mid x \leq y \,\}$.

Nechť $U[x]$ je seznam reprezentující množinu $u(x) = \{\ y \in P \mid x \prec y\ \}$. Nechť množina P je sjednocením po dvou disjunktních řetězců $C_1, \ldots, C_k$, nechť $C[x] = j$ značí, že $x \in C_j$.

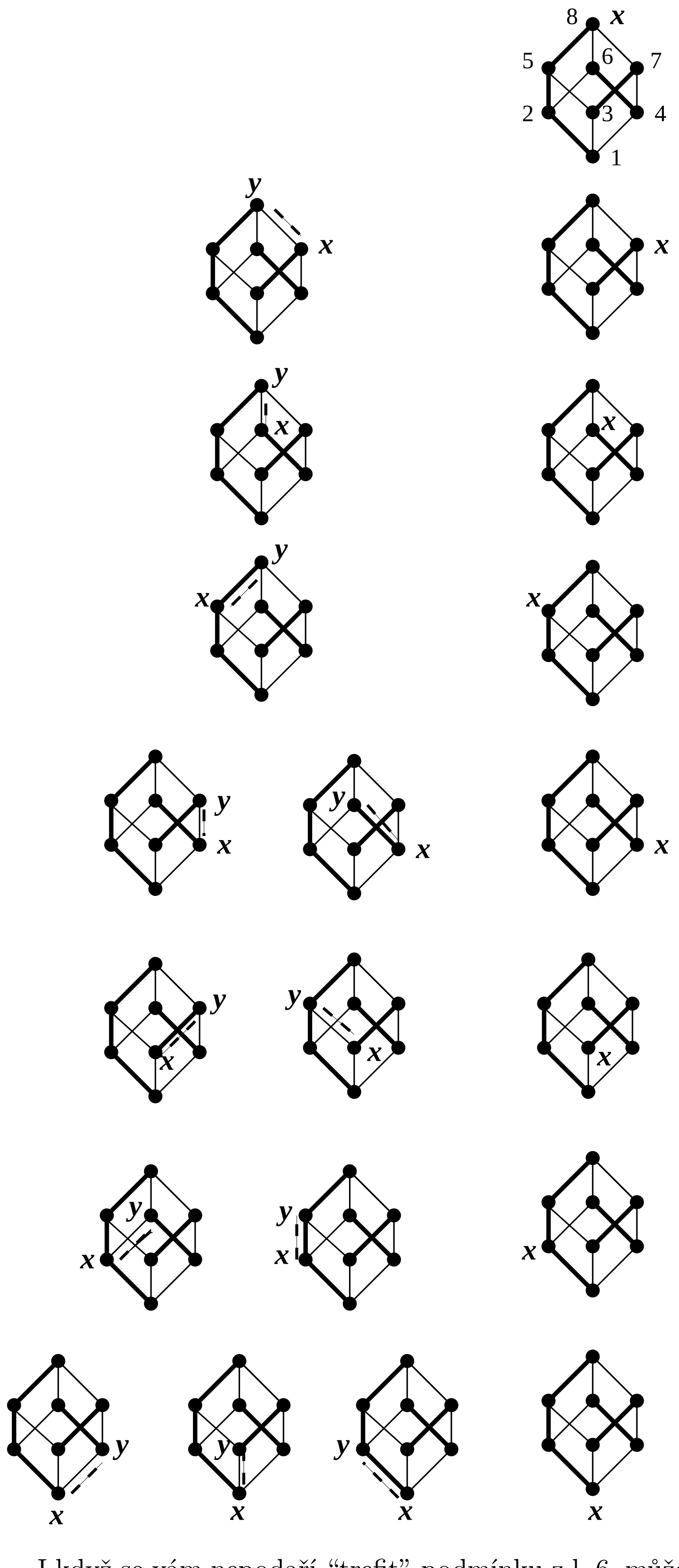
První položku seznamu S značíme $first\, S$.

Doplňte následující algoritmus. Dokažte jeho korektnost a odhadněte jeho složitost vzhledem k parametrům n, k, c, o.

1 invertujeme seznam P, nechť vznikne seznam S
2 **for** $x \in S,\ j = 1, \ldots, k$ **do** $G[j, x] \longleftarrow \emptyset$ **endfor**
3 **for** $x \in S$ **do**
4   **for** $y \in U[x]$ **do**
5   **for** $j = 1, \ldots, k$ **do**
6   **if** .... **then**
7         $G[j, x] \longleftarrow G[j, y]$
8   **endif**
9   **endfor**
10   **endfor**
11   $push(x, G[C[x], x])$
12 **endfor**
13 **for** $x \in S$ **do** $F[x] \longleftarrow \emptyset$ **endfor**
14 **for** $x \in S$ **do**
15   **for** $j = 1, \ldots, k$ **do**
16     $F[x] \longleftarrow G[j, x] \cup F[x]$
17   **endfor**
18 **endfor**

Nechť $g(j, x)$ je jistá podmnožina množiny C_j reprezentovaná seznamem $G[j, x]$.

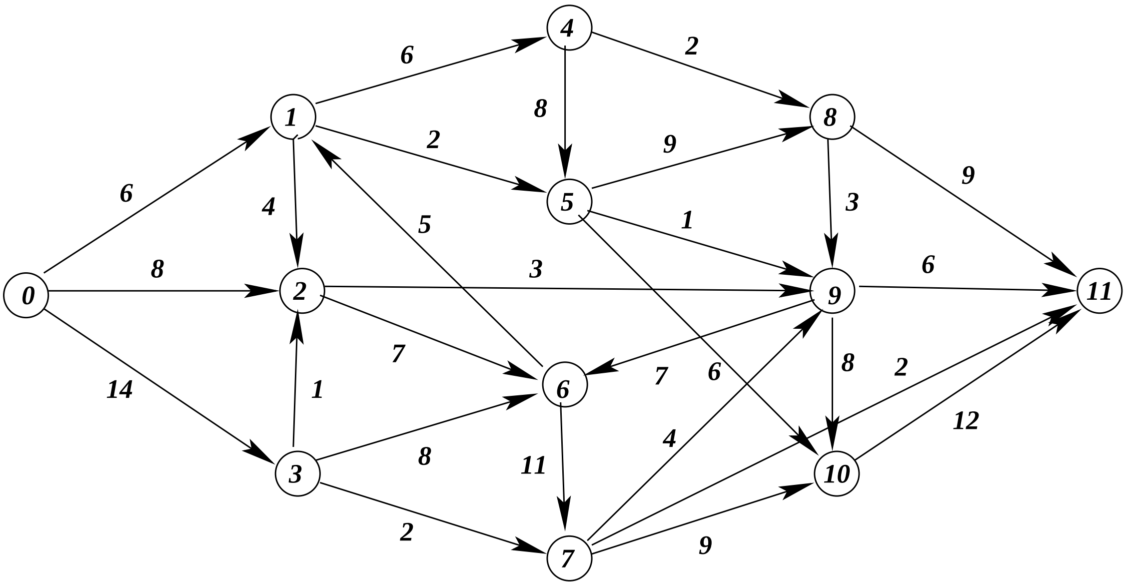
Na připravených diagramech uveďte stav množin $g(j, x)$ v okamžicích l. 10 a l. 12. Máme $C_1 = \{1, 2, 5, 8\},\ C_2 = \{3, 7\},\ C_3 = \{4, 6\}$ a topologické uspořádání množiny P je $(1, 2, \ldots, 8)$.

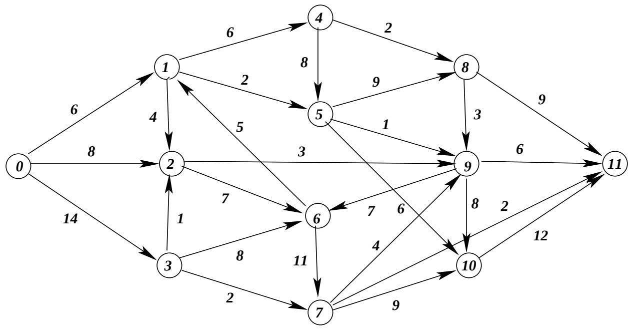


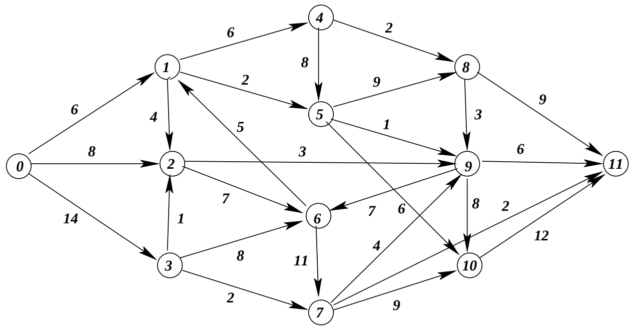
I když se vám nepodaří “trefit” podmínku z l. 6, můžete dostat body za odhad složitosti. Předpokládejte, že její ověření vezme konstantní čas.

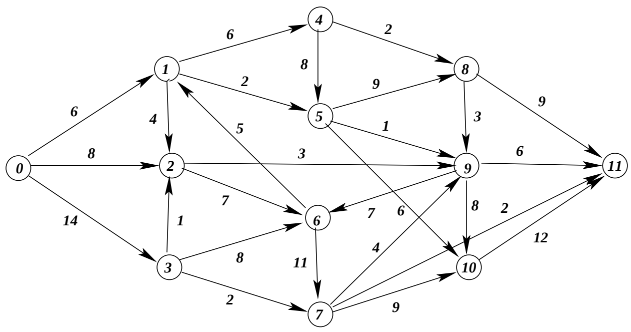
2. Toky v sítích

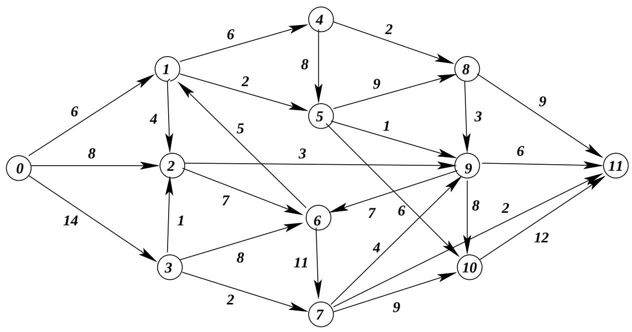
Na síti z obrázku se zdrojem 0 a cílem 11 najděte maximální tok. Podejte důkaz jeho maximálnosti.

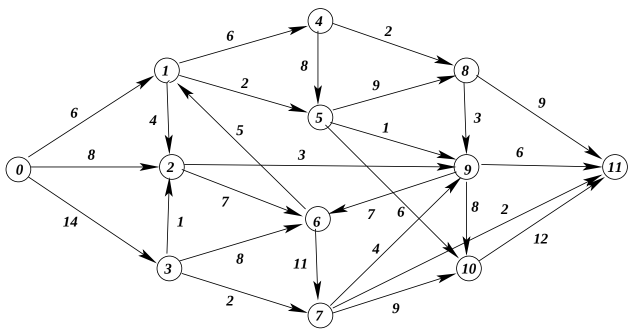


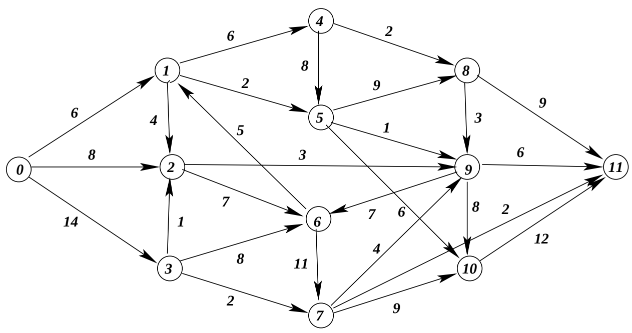












GRAFOVÉ ALGORITMY

2004/2005 - 2. termín

1. Rozklad uspořádané množiny na řetězce.

Konečná uspořádaná množina $\mathcal{P} = (P, \leq)$ je zadána pomocí relace pokrývání $\prec$. Nechť $U[x]$ je seznam reprezentující množinu $u(x) = \{ y \in P \mid x \prec y \}$. Nechť $n = |P|,\ c = |\prec|$. Nechť množina P je reprezentována seznamem odpovídajícím nějakému topologickému uspořádání grafu $(P, \prec)$.

Pro $x \in P$ máme spočítat $C[x] \in \{1, \ldots, k\}$ dávající rozklad množiny P na neprázdné řetězce $C_j = \{ x \in P \mid C[x] = j \},\ j \in \{1, \ldots, k\}$.

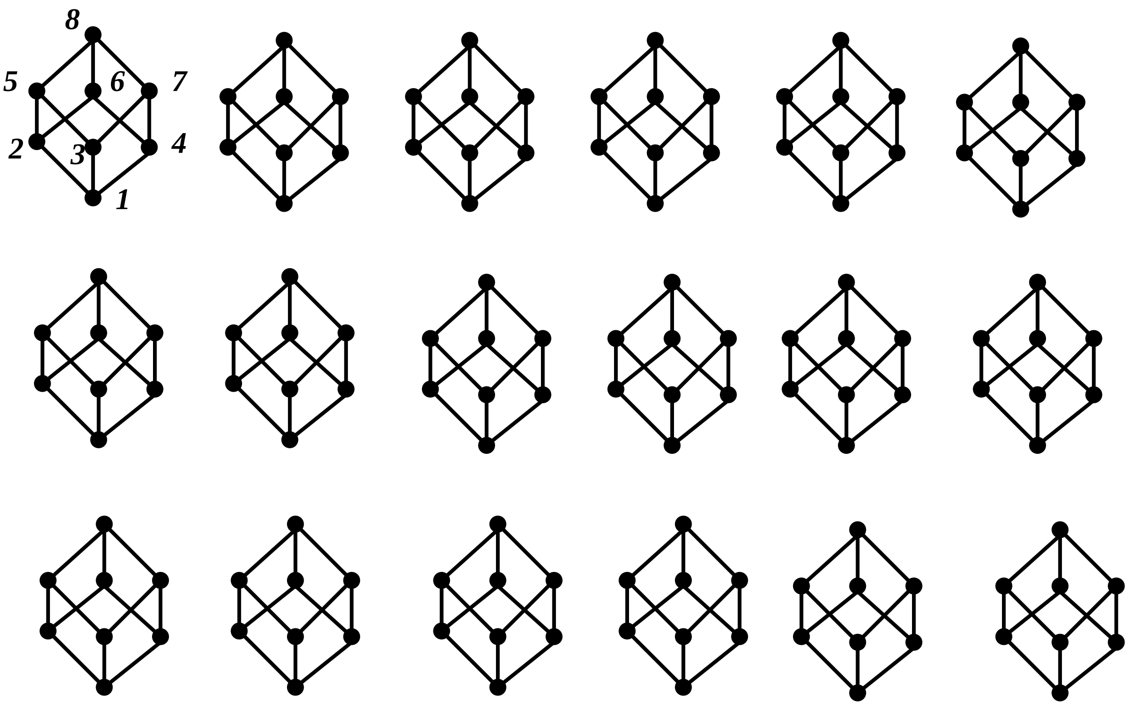
Pro $x \in P$ je význam $Z[x] \in \{0, 1\}$ "nezařazeno"/"zařazeno". Nechť množina l "kandidátů" je reprezentována seznamem L.

První položku seznamu S značíme $first\, S$. Neprázdný seznam S ochuzený o první položku značíme $rest\, S$.

Doplňte následující algoritmus.

```
1  j ⟵ 1
2  for x ∈ P do Z[x] ⟵ 0 endfor
3  for x ∈ P do
4    if Z[x] = 0 then
5      C[x] ⟵ j
6      Z[x] ⟵ 1
7      L ⟵ U[x]
8      while L ≠ ∅ do
9        y ⟵ first L
10         if Z[y] = 0 do
11
12
13
13'
14         else
15
16         endif
17       endwhile
18       j ⟵ j + 1
19   endif
20 endfor
```

Na připravených diagramech uveďte každou změnu množiny l společně s číslem řádku, na němž ke změně došlo. Topologické uspořádání množiny P je $(1, 2, \ldots, 8)$. Seznamy $U[x]$ respektují toto uspořádání.



Dokažte jeho korektnost. Důkaz může vypadat např. takto :
1. Každé C_j je neprázdný řetězec :

2. $\bigcup_{j=1,\dots,k} C_j = P$:

3. Pro $j \neq j'$ je ....

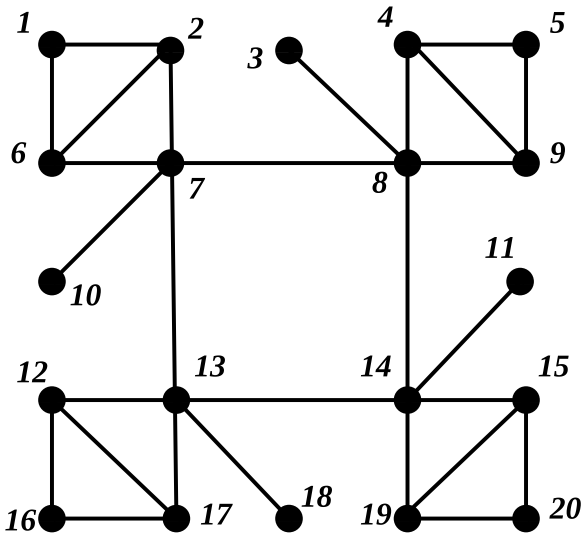
Bedlivě odhadněte složitost vašeho algoritmu vzhledem k parametrům n, c. Odhadněte též c pomocí n a uveďte složitost algoritmu vzhledem k n. I když se vám nepodaří "trefit" příkazy z l. ...., můžete dostat body za odhad složitosti. Předpokládejte, že každý z těchto příkazů vezme konstantní čas.

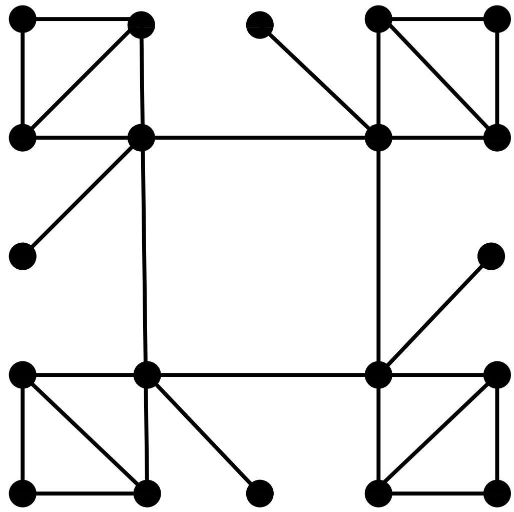
Konečně poznamenejte, co by se stalo, kdyby množina P nebyla topologicky uspořádaná.

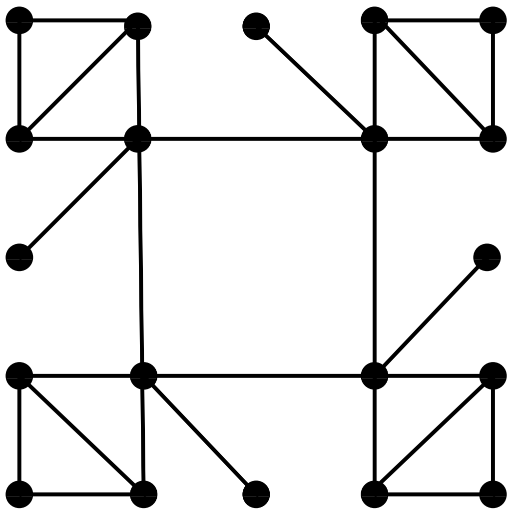
2. Párování.

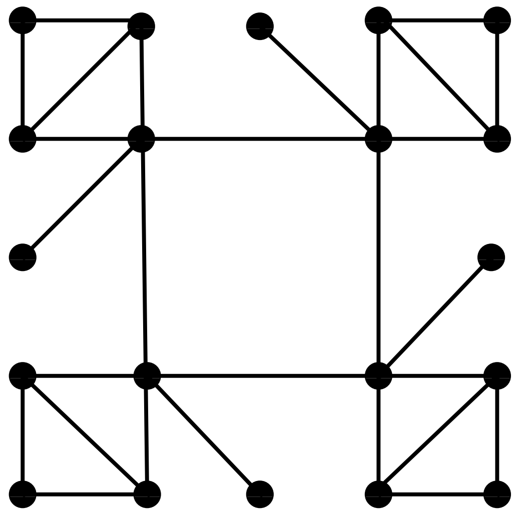
a) Ukažte, že níže uvedený graf $G = (\{1, \ldots, 20\}, \ldots)$ není bipartitní.

b) Najděte v našem grafu maximální párování.









GRAFOVÉ ALGORITMY

2004/2005 - 3. termín

1. Výpočet relace pokrývání.

Konečná uspořádaná množina $\mathcal{P} = (P, \leq)$ je zadána pomocí matice určující relaci $\leq$ (pro $x, y \in P$ v konstantním čase zjistíme zda $x \leq y$ či nikoliv). Sama množina P je reprezentována seznamem odpovídajícímu nějakému topologickému uspořádání grafu $(P, \leq)$.

Úkolem je spočítat seznamy $U[x]$, $x \in P$, reprezentující množiny

$$u(x) = \{\, y \in P \mid x \prec y \,\} \; .$$

První položku seznamu S značíme $first\, S$. Neprázdný seznam S ochuzený o první položku značíme $rest\, S$. Doplňte následující algoritmus.

```
Q ⟵ P
for a ∈ P
  Q ⟵ rest Q
  S ⟵ ∅
  for x ∈ Q do
    if a ≤ x then
      T ⟵ S
      while T ≠ ∅ and first T............. do
        .....................................
      endwhile
      if T = ∅ then push (x, S) endif
    endif
  endfor
  U[a] ⟵ S
endfor
```

Konkrétní případ : Matice relace $\leq$ je níže (na pozici (x, y) je 1 právě když $x \leq y$). Topologické uspořádání je $(1, 2, \ldots, 8)$.

	1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	1	1
2	0	1	0	0	1	1	0	1
3	0	0	1	0	1	0	1	1
4	0	0	0	1	0	1	1	1
5	0	0	0	0	1	0	0	1
6	0	0	0	0	0	1	0	1
7	0	0	0	0	0	0	1	1
8	0	0	0	0	0	0	0	1

Do připravené tabulky uveďte všechny změny proměnných x, S, T v části výpočtu, kde $a = 1$.

řádek	x	S	T

Dokažte jeho korektnost. Důkaz může vypadat např. takto :
1. Do seznamu S se může dostat jen prvek pokrývající prvek a.

2. Libovolný prvek zařazovaný do S pokrývá a.

Nechť $n = |P|$, $c = |\leq|$. Odhadněte složitost našeho algoritmu. I když se vám nepodaří trefit" formule/příkazy z l. ..., můžete dostat body za odhad složitosti. Předpokládejte, že každý z těchto příkazů vezme konstantní čas.

2. Cirkulace

Nechť $V = \{1, \ldots, n\}$ a nechť $G = (V, E)$ je orientovaný graf s ohodnocením hran $c : E \to \mathbb{R}$. V konkrétním příkladě uvedeném níže najděte maximální cirkulaci. Například můžete použít algoritmus:

Především každá cirkulace x určuje pomocný graf $G_x = (V, E_x)$ s ohodnocením hran w_x:

$$(i, j) \in E_x \iff (\ (i, j) \in E \ \&\ x_{ij} < c_{ij} \text{ nebo to neplatí a } (j, i) \in E \ \&\ x_{ji} > 0\ )\ .$$

Hrany prvního typu ohodnotíme -1 a hrany druhého typu 1.

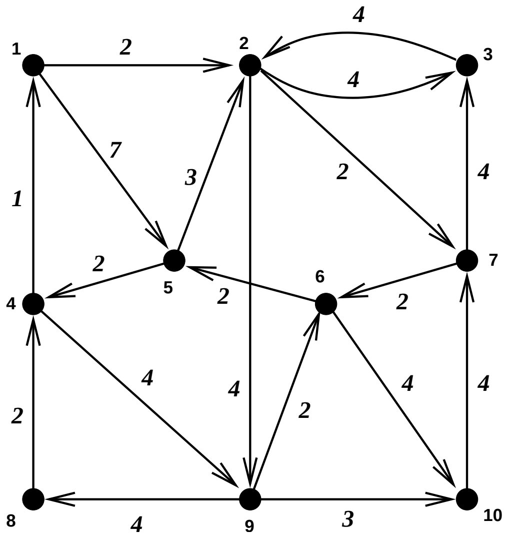
1. Začínáme s nulovou cirkulací.

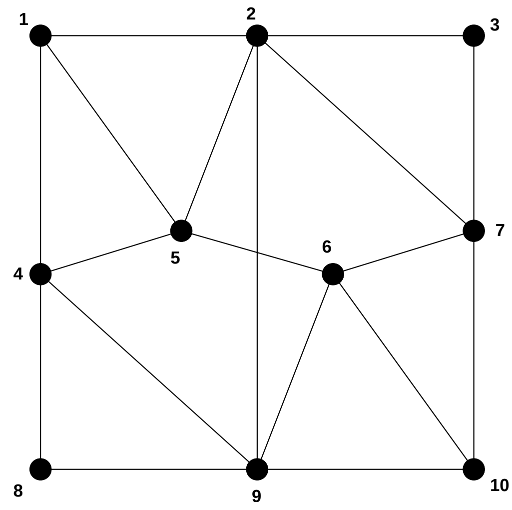
2. `while` v (G_x, w_x) existuje záporná kružnice

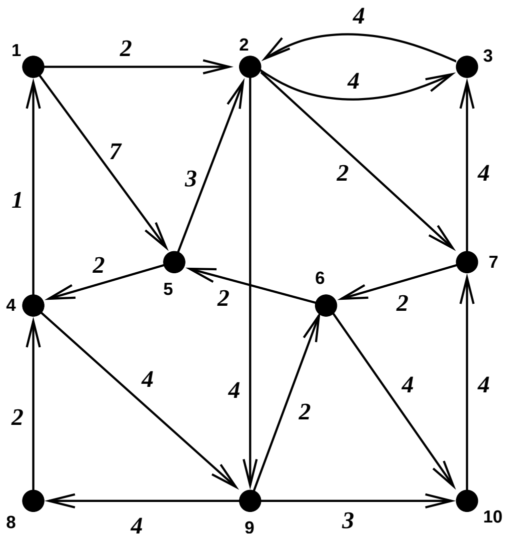
`do` vybereme některou a zvětšíme podél ní cirkulaci.

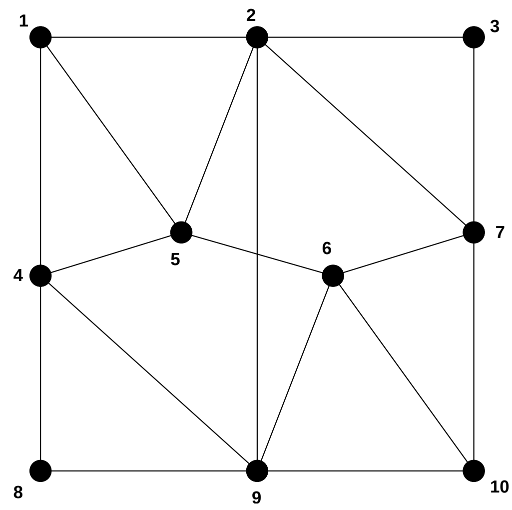
Abyste se ve výpočtech vyznali, doporučuji např. cirkulace modře, hrany pomocného grafu s ohodnocením -1 zeleně, ty s ohodnocením 1 červeně, další barvu můžete použít pro záporné kružnice.

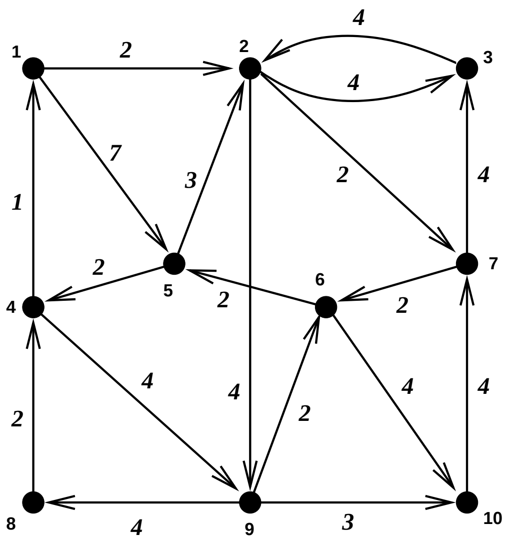
Hodnocení: nalezení maximální cirkulace 10 b. Důkaz maximality (např. pomocí některého algoritmu pro hledání záporných kružnic) dalších 10 b.

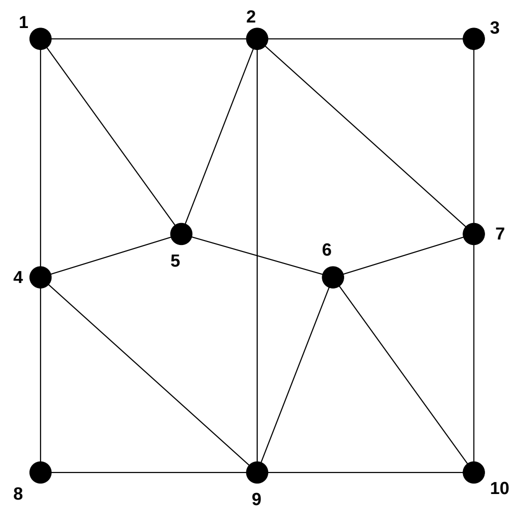

4
1
2
2
3
4
7
3
2
4
1
2
5
2
6
2
4
4
7
4
4
2
2
4
8
4
9
3
10

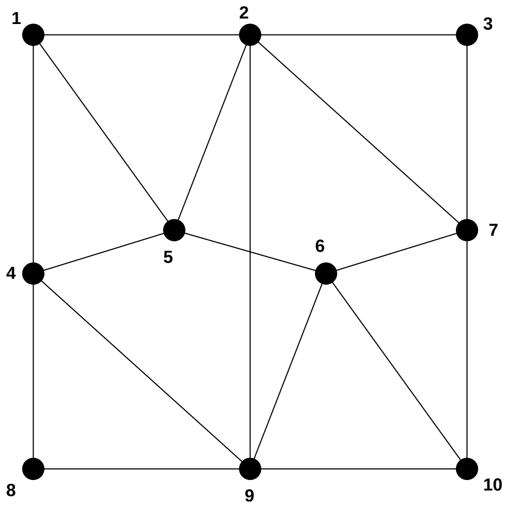
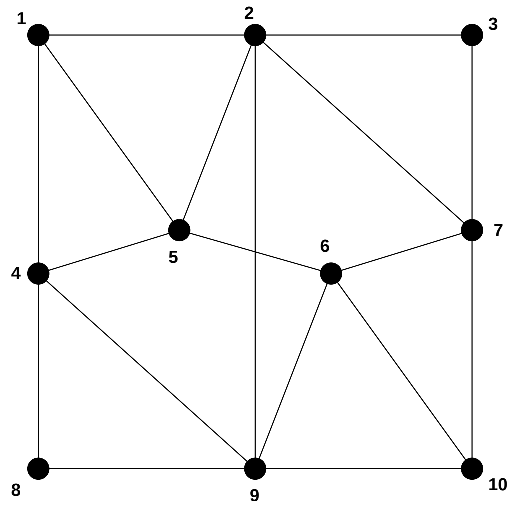
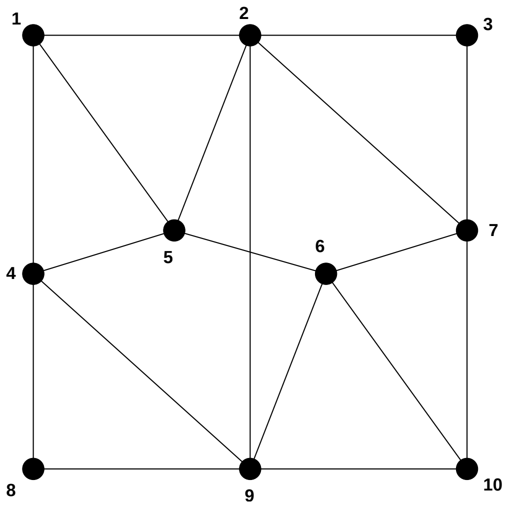
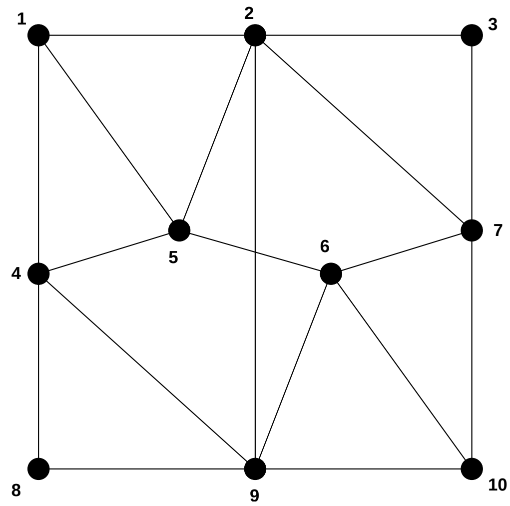
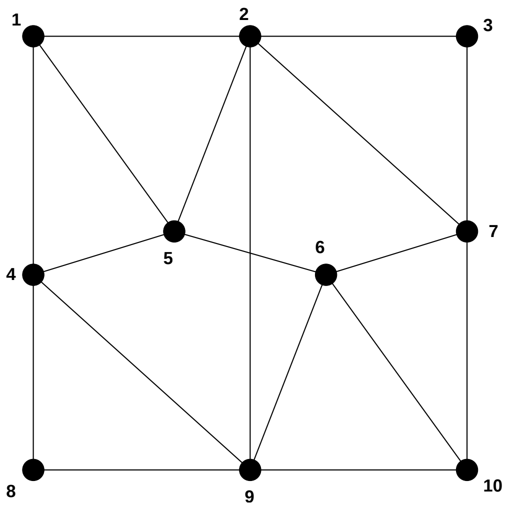
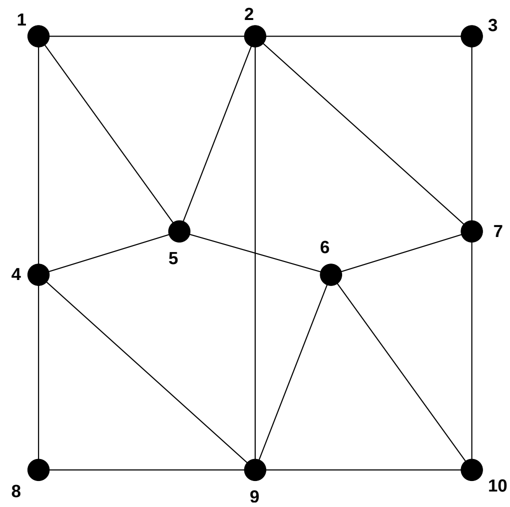

1
2
3
7
4
5
6
8
9
10


4
1
2
2
3
4
7
3
2
4
1
2
5
2
6
2
4
4
7
4
4
2
2
4
8
4
9
3
10


1
2
3
7
4
5
6
8
9
10


4
1
2
2
3
4
7
3
2
4
1
2
5
2
6
2
4
4
7
4
4
2
2
4
8
4
9
3
10


1
2
3
7
4
5
6
8
9
10


1
2
3
4
5
6
7
8
9
10

1
2
3
4
5
6
7
8
9
10

1
2
3
4
5
6
7
8
9
10

1
2
3
4
5
6
7
8
9
10

1
2
3
4
5
6
7
8
9
10

1
2
3
4
5
6
7
8
9
10