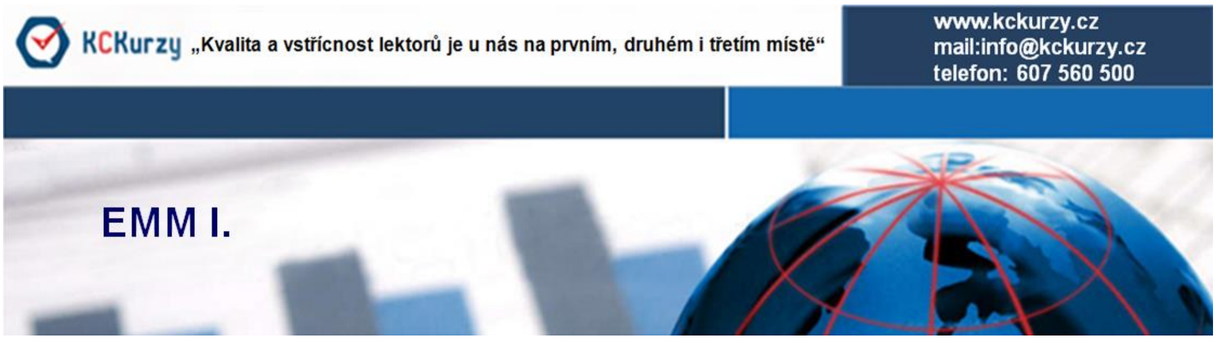


WWW.KCKURZY.CZ
Jak udělat zkoušku z EMM1 na ČZU

Vážení přátelé matematici,
v tomto materiálu najdete soubor typových písemkových příkladů z EMM1 na ČZU, podle kterých vysvětluji EMM1 na našich pravidelných lekcích a nalejvárnách před zkouškou. Výběr příkladů je zvolen tak, že pokrývá většinu typů, které se u zkoušky objevují. Přihlásit se můžete kdykoliv na **www.kckurzy.cz.**

Pokud budete ovládat všechny příklady z tohoto souboru, ze svých zkušeností si dovoluji tvrdit, že zkoušku velmi pravděpodobně uděláte.

Přeji příjemné učení☺. RNDr. Marian Rybář – lektor matematiky

Přehled probíraných témat:

1. Lineární programování (LP) - prostor řešení
2. Lineární programování (LP) - prostor požadavků
3. Simplexový algoritmus + 4. Rozbor řešení
5. Postoptimalizační analýza simplexové tabulky
6. Jednostupňová dopravní úloha (JDÚ) – výchozí řešení
7. Jednostupňová dopravní úloha (JDÚ) – optimalizace
8. Dvoustupňová dopravní úloha (DDÚ)
9. Další dopravní úlohy
   9.1 Přiřazovací úloha (Maďarská metoda)
   9.2 Jednookruhový okružní problém - Metoda nejbližšího souseda
   9.3 Jednookruhový okružní problém - Metoda VAM
   9.4 Víceokruhový okružní problém (Mayerova metoda)
10. Teorie grafů
   10.1 Minimální kostra
   10.2 Nejkratší cesta v grafu (Dijkstrův algoritmus)
   10.3 Maximální tok v síti (Ford-Fulkersonův algoritmus)
11. Projektové řízení – metoda CPM

1. Lineární programování (LP) - prostor řešení

Tento způsob používáme v modelech, které mají pouze 2 proměnné x1, x2 a neomezený počet omezujících podmínek.

Příklad 1:

Management podniku musí vyřešit následující problém:
Na ploše 14 ha je třeba pěstovat plodiny A a B tak, aby se dosáhlo maximálního zisku. Je požadováno vyrobit alespoň 32 tun krmných jednotek. K dispozici je celkem 100 hodin lidské práce. Plodinu A lze pěstovat nejvýše na 11 ha. Ostatní výrobní faktory nejsou limitovány. Další informace jsou uvedeny v tabulce:

Na 1 ha	**Plodina**	
	A (žito)	**B (pšenice)**
Výnos krmných jednotek (t)	4	2
Potřeba hodin lidské práce (h)	5	10
Zisk z jednoho hektaru plodiny (tis. Kč)	3	2

Kuchařka Příklad 1:

Krok 1: Označím si neznámé proměnné

X1…….hledaná optimální plocha plodiny A (žita) v ha
X2……. hledaná optimální plocha plodiny B (pšenice) v ha
Z …….. celkový zisk

Krok 2: Slovní zadání postupně přepíšu větu po větě do rovnic a nerovnic (formulace modelu lineárního programování)

3 části:

- **Účelová funkce** (to co chci maximalizovat či minimalizovat)
  Zmax = 3.x1+2.x2

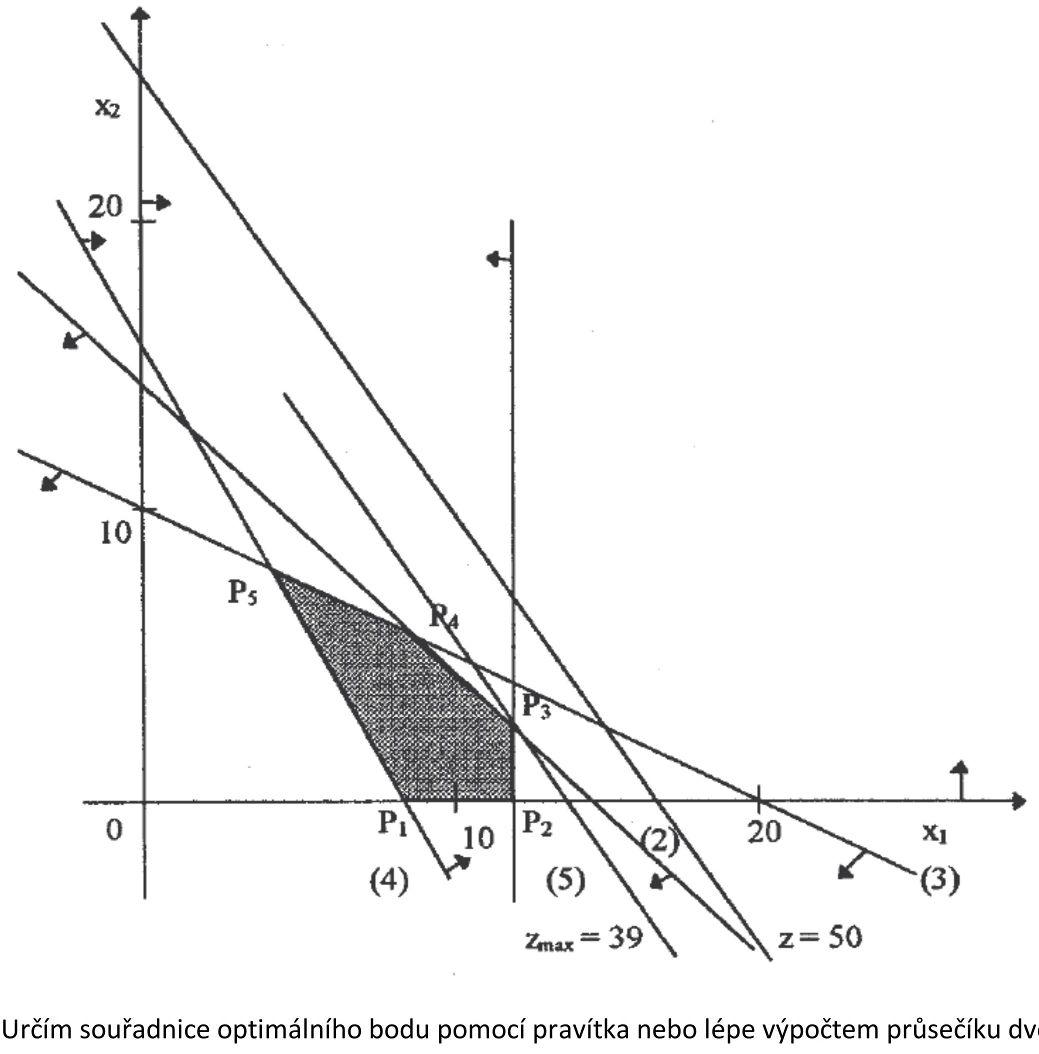
- **Omezující podmínky** (to co mně omezuje)
  x1+x2 ≤ 14 (celková plocha pole)
  4.x1+2.x2 ≥ 32 (minimální požadovaný počet krmných jednotek)
  5.x1+10.x2 ≤ 100 (lidská práce)
  x1 ≤ 11 (omezení pro plodinu A)

- **Podmínky nezápornosti** (můžeme pěstovat pouze kladná množství žita a pšenice)
  X1 ≥ 0
  X2 ≥ 0

Krok 3:

a) Zakreslím graficky množinu omezujících podmínek do **prostoru řešení** (vznikne tzv. konvexní polyedr = simplex), což jsou kombinace množství plodin A a B (x1 a x2), kde je pěstování z důvodů omezujících podmínek možné (šedá barva) – podrobný nákres viz doučování.
Pozn: Optimální bod čekáme v případě maximalizace někde „vpravo a nahoře" – čím víc pěstuji, tím mám větší zisk.

b) Zakreslím jinou barvou graf účelové funkce (přímku) pro nějaký libovolný zisk např. z = 50. Posunu rovnoběžku s touto „pomocnou" účelovou funkcí až do bodu, kde se co nejdále od počátku „vpravo a nahoře" (v případě maximalizace) dotkne konvexního polyedru. V tomto místě je optimální bod, který mi určí optimální kombinaci množství plodin x1 a x2, při kterém bude maximální zisk (bod P3).



Krok 4: Určím souřadnice optimálního bodu pomocí pravítka nebo lépe výpočtem průsečíku dvou funkcí, jejichž je průsečíkem

X1 = 11
x1 + x2 = 14
………………………
Tedy x1 = 11, x2 = 3
P3 = [11;3]

Krok 5: Dosadíme výsledné optimální hodnoty x1 a x2 do účelové funkce Zmax = 3.x1+2.x2 a dopočítáme výsledný maximální zisk

Zmax = 3.11+2.3 = 39 tis. Kč

Krok 6: **Slovní odpověď**: Optimální pěstované množství plodiny A (žita) je 11 ha. Optimální pěstované množství plodiny B (pšenice) je 3 ha. Zisk bude 39 tis. Kč.

Domácí úkol☺

Příklad 2:

Podnik vyrábí výrobky A a B, které je třeba opracovat na dvou strojích. Doba provozu strojů je omezená. Požadavky na opracování a pracovní doba strojů jsou v tabulce. Stanovte takovou strukturu výroby, při níž by prodejem výrobků bylo dosaženo maximálního zisku.

	Požadavek na čas opracování na prvním stroji (hod)	Požadavek na čas opracování na druhém stroji (hod)	Zisk Kč/kus
VýrobekA	2	2	6
VýrobekB	3	1	7
Celkový disponibilní čas na prvním stroji			24 hod
Celkový disponibilní čas na druhém stroji			16 hod

Řešení: Slovní odpověď: Optimální počet výrobků A bude 6. Optimální počet výrobků B bude 4. Zisk bude 64 tis. Kč.

2. Lineární programování (LP) - prostor požadavků

Tento způsob používáme v modelech, které mají neomezený počet proměnných x1, x2, x3, x4, …, ale pouze 2 omezující podmínky.

Příklad 1:

Zemědělec má k dispozici čtyři druhy krmiv: krmný ječmen, zelenou píci, seno víceletých pícnin a kukuřičnou siláž. Sestavuje krmnou dávku tak, aby obsahovala nejméně 5 škrobových jednotek (ŠJ) a nejvýše 850 jednotek stravitelných dusíkatých látek (SNL).

Živiny v krmivu	Obsah ŠJ (kg/kg)	Obsah SNL (g/kg)	Cena (Kč/kg)
Krmný ječmen (x1)	0,75	75	3
Zelená píce (x2)	0,1	20	0,4
Seno VLP (x3)	0,4	70	1
Kukuřičná siláž (x4)	0,1	15	0,5

Jaká krmiva a v jakém množství má zemědělec použít, aby byla krmná dávka co nejlevnější? Jaká bude cena této krmné dávky?

Kuchařka Příklad 1:

Krok 1: Označíme si neznámé proměnné

X1…….krmný ječmen (kg)
X2…….zelená píce (kg)
X3…….seno VLP (kg)
X4…….kukuřičná siláž (kg)
Z …….. cena směsi v Kč

Krok 2: Slovní zadání přepíšeme větu po větě do rovnic a nerovnic (formulace modelu lineárního programování)

3 části:

- **Účelová funkce** (to co chci maximalizovat či minimalizovat)
  Zmin = 3.x1+0,4.x2+x3+0,5x4
- **Omezující podmínky** (to co mně omezuje)
  0,75.x1+0,1.x2+0,4.x3+0,1.x4 ≥ 5 (kg ŠJ)
  75.x1+20.x2+70.x3+15.x4 ≤ 850 (g SNL)
- **Podmínky nezápornosti** (můžeme pěstovat pouze kladná množství žita a pšenice)
  X1,2,3,4 ≥ 0

Krok 3: Pomocí tzv. **doplňkových proměnných d1, d2** převedeme nerovnice na rovnice a v účelové funkci dáme před všechny doplňkové proměnné 0

3 části:

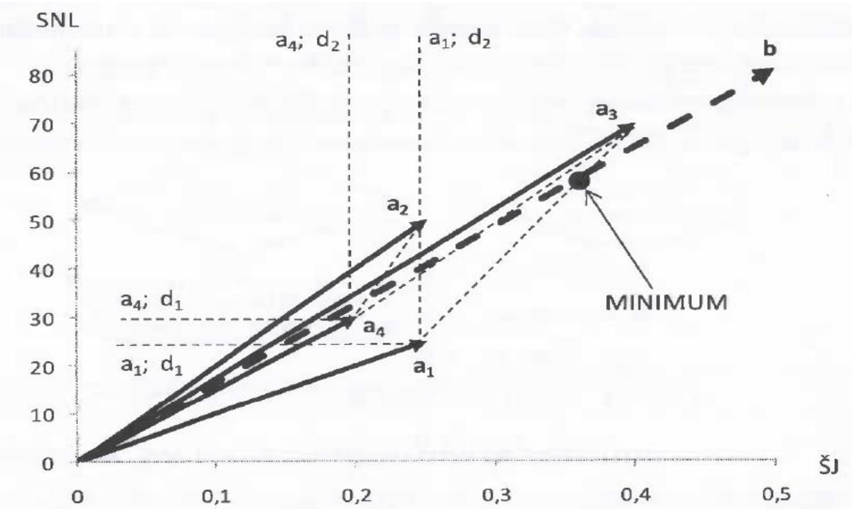
- **Účelová funkce**
  Zmin = 3.x1+0,4.x2+x3+0,5x4+0.d1+0.d2
- **Omezující podmínky**
  0,75.x1+0,1.x2+0,4.x3+0,1.x4 –1.d1 = 5 (kg ŠJ)
  75.x1+20.x2+70.x3+15.x4 +1.d2 = 850 (g SNL)
- **Podmínky nezápornosti** (můžeme pěstovat pouze kladná množství)
  X1,2,3,4, d1, d2 ≥ 0

Krok 4: Vydělíme koeficienty v obou rovnicích omezujících podmínek cenami z účelové funkce a získáme pro každou proměnnou x1, x2, x3 a x4 jeden vektor a1, a2, a3 a a4 (např. pro proměnnou x1 vektor a1 atd). Ukázka viz doučování.

Získáme vektory a1 = $\binom{0{,}25}{25}$, a2 = $\binom{0{,}25}{50}$, a3 = $\binom{0{,}4}{70}$, a4 = $\binom{0{,}2}{30}$, vektory d1 = $\binom{-\infty}{0}$, d2 $= \binom{0}{\infty}$

a vektor pravých stran b (pouze opíšeme pravé strany omezujících podmínek) b = $\binom{5}{850}$

Krok 5: Zakreslíme výše vypočítané vektory a1, a2, a3, a4, d1, d2 a vektor pravých stran b do **prostoru požadavků** (v našem případě máme požadavky ŠJ a SNL z omezujících podmínek). Obrázek viz ukázka na doučování.



V případě **minimalizace** hledáme spojnici mezi konci vektorů „a“ a „d“, která „nejdále vpravo nahoře“ protne vektor b. V případě **maximalizace** hledáme spojnici mezi konci vektorů „a“ a „d“, která „nejblíže vlevo dole“ protne vektor b. Viz ukázka doučování.

V našem případě je řešením spojnice mezi vektory a1 a a3. Optimální krmná dávka bude tedy složena z krmného ječmene (x1) a sena VLP (x3).

Krok 6:
V rovnicích omezujících podmínek z Kroku 3 necháme pouze proměnné x1 a x3 a vyřešíme vzniklou soustavu 2 rovnic o dvou neznámých:
0,75.x1 +0,4.x3 = 5
75.x1 +70.x3 = 850
Řešením je X1 = 0,44, X3 = 11,67

Krok 7: Dosadíme výsledné optimální hodnoty x1 a x3 do účelové funkce Zmin = 3.x1+0,4.x2+x3+0,5x4+0.d1+0.d2 a dopočítáme výslednou minimální cenu směsi
Zmin = 3.0,44+11,67 = 13 Kč

Krok 8: **Slovní odpověď**: Nejlevnější krmnou dávku, která splní požadavky ze zadání, tedy sestavíme z 0,44 kg krmného ječmene (x1) a 11,67 kg sena VLP (x3). Cena krmné dávky bude 13 Kč.

Domácí úkol☺

Příklad 2:

Je dán model lineárního programování
$4x1 + 0,5x2 + 4x3 + 4x4 \geq 4$
$x1 + 3x2 + 2x3 \leq 3$
$z\ max = 4x1 + x2 + 2x3 + 6x4$
$x1, x2, x3, x4 \geq 0$

a) Za jakých podmínek je možné model lineárního programován řešit graficky? (3 b)
b) Vyřešte tento model graficky pomocí vhodného zobrazení. (12 b)
c) Z výsledku grafického řešení odvoďte hodnoty všech proměnných v optimálním řešení a hodnotu účelové funkce. (10 b)

Řešení: x1 = 3, d1 = 8, z max = 12

3. Simplexový algoritmus + 4. Rozbor řešení

Příklad 1:

Pěstujeme pšenici (x1), žito (x2) a ječmen (x3). Pšenici je možno pěstovat maximálně na 2 ha pole. Na 1 ha pšenice je potřeba 1 den práce, na žito 1 den práce a na ječmen 2 dny práce. K dispozici jsou maximálně 4 dny práce. Na pěstování pšenice není potřeba hnojivo, na 1 ha žita je potřeba 3 tuny hnojiva a na 1 ha ječmene je potřeba 4 tuny hnojiva. K dispozici je maximálně 6 tun hnojiva. Z 1 ha pšenice máme zisk 1 tis. Kč, z 1 ha žita máme zisk 2 tis. Kč a z 1 ha ječmene máme zisk 4 tis. Kč. Máme určit plochu pšenice, žita a ječmene aby zisk byl maximální.

Kuchařka Příklad 1:

Krok 1: Označím si neznámé proměnné

X1…….hledaná optimální plocha pšenice v ha
X2……. hledaná optimální plocha žita v ha
X3……. hledaná optimální plocha ječmene v ha
Z……... celkový zisk

(X1–X3:) **tzv. rozhodovací (strukturní) proměnné**

Krok 2: Slovní zadání postupně přepíšu větu po větě do rovnic a nerovnic (formulace modelu lineárního programování)

3 části:

- **Účelová funkce** (to co chci maximalizovat či minimalizovat)
  Zmax = x1+2.x2+4.x3

- **Omezující podmínky** (to co mně omezuje)
  x1 ≤ 2 (omezení plochy pro pšenici)
  x1+x2+2.x3 ≤ 4 (omezená lidská práce)
  3.x2+4.x3 ≤ 6 (omezené hnojivo)

- **Podmínky nezápornosti** (můžeme pěstovat pouze kladná množství žita a pšenice)
  X1 ≥ 0, X2 ≥ 0, X3 ≥ 0

Krok 3: Pomocí tzv. **doplňkových proměnných x4, x5, x6** převedeme nerovnice na rovnice a v účelové funkci dáme před všechny doplňkové proměnné 0 (doplňkové proměnné mají význam nevyužité plochy – vydělám na nich 0 tis. Kč)

x1 + x4 = 2
x1+x2+2.x3 + x5 = 4
3.x2+4.x3 + x6 = 6
Zmax = 1.x1 + 2.x2 + 4.x3 + 0.x4 + 0.x5 + 0.x6

Krok 4: **Sestavení simplexové tabulky**

a) Do tabulky nahoru napíšeme všechny proměnné x1, …, x6 a nad ně do horního řádku dáme koeficienty (čísla), které byly před nimi v účelové funkci.
b) Pod to do každého řádku přepíšeme jednu omezující podmínku (pouze čísla) včetně pravých stran (v našem případě 2,4,6).
c) Před tabulku zleva do sloupce Z dáme proměnné, které jsou v té chvíli v bázi (proměnné, pod kterými jsou v tabulce jednotkové vektory).
d) Do sloupce ci dáme koeficienty, které jsou před bázovými proměnnými v účelové funkci (tedy i v tabulce nad bázovými proměnnými).
e) Vypočteme řádek zj-cj pod tabulkou: vynásobíme vždy mezi sebou postupně všechna čísla ze sloupce ci se všemi čísly ve sloupci pod danou proměnnou x1, x2, x3, atd a odečteme od výsledku číslo z horního řádku. Výsledek napíšeme dolů do řádku zj-cj pod danou proměnnou.

c_i $i \in Z$	Z	1	2	4	0	0	0	x_i $i \in Z$	θ_{min}
		x_1	x_2	x_3	x_4	x_5	x_6		
0	x_4	1	0	0	1	0	0	2	-
0	x_5	1	1	2	0	1	0	4	4/2
0	x_6	0	3	4	0	0	1	6	6/4
$z_j - c_j$		−1	−2	−4	0	0	0	0	

Zisk = 0.2 + 0.4 + 0.6 = 0

f) V řádku zj-cj pod tabulkou vidíme záporná čísla. V případě maximalizace to znamená, že řešení ještě není optimální (**tzv. test optimality**). Musíme pokračovat dále.

Krok 5: **Simplexový algoritmus**

a) Vyberu z posledního řádku zj-cj nejvíce záporný prvek (pokud se jedná o maximalizaci), v našem případě -4. Pokud se jedná o minimalizaci, bereme nejvíce kladný prvek.
b) Toto číslo mi určí tzv. **klíčový sloupec**. Ten ukazuje na proměnnou x3, která je nejvíce výhodná a zanedlouho vstoupí do **báze** (do výběru výhodných proměnných).
c) Dále dělíme čísla pravých stran z předposledního sloupce xi čísly z klíčového sloupce a výsledky zapisujeme do posledního sloupce Ɵmin (**tzv. test přípustnosti**). Nejmenší z těchto výsledků dělení mi určí tzv. **klíčový řádek.** Ten ukazuje na proměnnou x6, která je nejméně výhodná a zanedlouho vystoupí z báze (z výběru výhodných proměnných).
d) Na spojnici klíčového sloupce a klíčového řádku najdeme tzv. **klíčový prvek (pivot).**
e) Na místě klíčového prvku (pivota) musíme získat číslo 1 a na místě všech ostatních čísel v klíčovém sloupci čísla 0. Postupujeme podobně jako u Gaussovy eliminace, která se používá při úpravě matic **(mezivýpočty zapisujeme do tabulky jinou barvou – viz doučování).**
f) Celý 3. řádek s klíčovým prvkem musíme vydělit číslem 4, abychom dostali místo původního čísla novou hodnotu 1. Tím se změní i všechna čísla ve 3. řádku.
g) Dále potřebujeme vynulovat číslo 2 v klíčovém sloupci, takže celý nově vzniklý 3. řádek vynásobíme číslem -2 a přičteme ke 2. řádku staré matice.
h) V 1. řádku klíčového sloupce máme číslo 0, takže stačí opsat.

i) Vznikla nám nová tabulka, ve které dopočítáme opět řádek zj-cj. Vidíme, že ani nyní není ještě řešení optimální (nemá v řádku zj-cj pouze kladná čísla).

c_j $i \in Z$	Z	1	2	4	0	0	0	x_j $i \in Z$ =	θ_{min}
		x_1	x_2	x_3	x_4	x_5	x_6		
0	x_4	1	0	0	1	0	0	2	2/1=2
0	x_5	1	−1/2	0	0	1	−1/2	1	1/1=1
4	x_3	0	3/4	1	0	0	1/4	3/2	-
$z_j - c_j$		−1	1				1	6	

Zisk = 0.2 + 0.1 + 4 . 3/2 = 6

j) Opakujeme kroky 5 a 6 tak dlouho, dokud v případě maximalizace v posledním řádku zj-cj nevzniknou pouze kladná čísla.

k) Získáme tabulku, ve které jsou v posledním řádku zj-cj už pouze kladná čísla, řešení je tedy optimální a simplexový algoritmus je tedy ukončen.

c_j $i \in Z$	Z	1	2	4	0	0	0	x_j $i \in Z$	θ
		x_1	x_2	x_3	x_4	x_5	x_6		
0	x_4	0	1/2	0	1	−1	1/2	1	
1	x_1	1	−1/2	0	0	1	−1/2	1	
4	x_3	0	3/4	1	0	0	1/4	3/2	
$z_j - c_j$			1/2			1	1/2	7	

Zisk = 0.1 + 1.1 + 4 . 3/2 = 7

Krok 6: **Sestavení a interpretace vektoru řešení**

a) Ve výsledném vektoru řešení budou pouze čísla ze sloupce xi odpovídající bázovým proměnným ze sloupce Z. U všech ostatních proměnných bude 0 – proměnné, které nejsou v bázi, nejsou pro maximalizaci zisku výhodné. Výsledný vektor řešení má tedy tvar:

X = (1; 0; 3/2; 1; 0; 0)

X1 x2 x3 x4 x5 x6

X1 = pšenici je optimální pěstovat na rozloze 1 ha

X2 = žito nebudeme pěstovat (z hlediska maximalizace zisku je nevýhodné)

X3 = ječmen je optimální pěstovat na rozloze 3/2 = 1,5 ha

X4 = nevyužitá plocha pro pěstování pšenice je 1 ha (pokud nám vyjde číslo u nějaké doplňkové proměnné, znamená to nevyužitou surovinu z rovnice, kde jsme tuto doplňkovou proměnnou dali)

Je úloha degenerovaná? Není, protože žádná z bazických proměnných (pravý sloupec) není nulová.
Má úloha alternativní řešení? Nemá, protože žádná z nebazických proměnných nemá v řádku zj-cj hodnotu 0.

Krok 7: **Dopočítám výsledný maximální zisk:** Z_{MAX} = 1.1 + 2.0 + 4.(3/2) + 0.1 + 0.1 + 0.0 = 7

Rozdíly v případě minimalizace:

a) Řešení je optimální (simplexový algoritmus končí) pokud všechna čísla v řádku zj-cj jsou ZÁPORNÁ
b) Klíčový sloupec vybíráme podle NEJVĚTŠÍHO čísla v dolním řádku zj-cj
c) Klíčový řádek vybíráme stejně jako u maximalizace podle NEJMENŠÍHO čísla v posledním sloupci tabulky ϴmin

Příklad 2:

Do výrobní linky je nutno zařadit **přesně** 10 přístrojů. Na trhu jsou dva typy A a B, které se liší cenou a produkcí. Přístrojů typu A musí být v lince alespoň 3 kusy. Celkové pořizovací náklady nesmějí přesáhnout 28. mil Kč. Kolik přístrojů zakoupíme a jak sestavíme výrobní linku, aby celková produkce byla maximální? Typ A stojí 4 mil. Kč, typ B stojí 1 mil. Kč, typ A produkuje výrobky v hodnotě 800 000 Kč za směnu, typ B v hodnotě 500 000 Kč za směnu.

Kuchařka Příklad 2:

Krok 1: Označím si neznámé proměnné **tzv. rozhodovací (strukturní) proměnné**

X1……. počet přístrojů typu A
X2……. počet přístrojů typu B
Z……… celkový zisk

Krok 2: Slovní zadání postupně přepíšu větu po větě do rovnic a nerovnic (formulace modelu lineárního programování)

3 části:

- **Účelová funkce** (to co chci maximalizovat či minimalizovat)
  Zmax =8x1+5.x2(volíme jednotku 100 000 Kč)
- **Omezující podmínky**
  X1+x2 = 10
  4.x1+x2 ≤ 28
  x1 ≥ 3
- **Podmínky nezápornosti**
  X1 ≥ 0, X2 ≥ 0

Krok 3: Pomocí tzv. **doplňkových proměnných x3, x4** převedeme nerovnice na rovnice a v účelové funkci dáme před všechny doplňkové proměnné 0 (doplňkové proměnné mají význam nevyužité suroviny – vydělám na nich 0 tis. Kč)

X1 + x2 = 10
4.x1+x2 + x3 = 28
x1 -x4 = 3
Zmax =8x1+5.x2+0.x3+0.x4
X1 ≥ 0, X2 ≥ 0, X3 ≥ 0, X4 ≥ 0

doplňkové proměnné

X3 = rezerva, nevyužité náklady

X4 = překročení počtu přístrojů A nad 3 ks

Krok 4: **Úprava modelu na tzv. Kanonický tvar**

Definice: Model je v **kanonickém tvaru**, pokud obsahuje tolik jednotkových vektorů, kolik má rovnic. Úpravu modelu na Kanonický tvar provedeme pomocí **pomocných (umělých) proměnných x5 a x6:**

x5 I x1 + x2 +x5 = 10

x3 I 4.x1 +x2 + x3 = 28

x6 I x1 -x4 +x6 = 3

Zmax =8x1+5.x2+0.x3+0.x4-100.x5-100.x6

Xj ≥ 0, j = 1,2,…,6

Před pomocné proměnné dáme tzv. **prohibitivní sazby**. V případě maximalizace např. M = -100. Volím extrémně nevýhodnou sazbu, aby bylo jasné, že pomocná proměnná nemůže být vybrána do báze (řešení). „Pokud mám na nějaké proměnné zisk -100, je jasné, že nebude vybrána do řešení“

Krok 5: **Sestavení a vyřešení simplexové tabulky**

c_i	Z	8	5	0	0	−100	−100	x_i	θ
$i \in Z$		x_1	x_2	x_3	x_4	x_5	x_6	$i \in Z$	
−100	x_5	1	1	0	0	1	0	10	10/1
0	x_3	4	1	1	0	0	0	28	28/4
−100	x_6	1	0	0	−1	0	1	3	3/1
$z_j - c_j$		−208	−105		100			*	
−100	x_5	0	1	0	1	1	−1	7	7/1
0	x_3	0	1	1	4	0	-4	16	16/4
8	x_1	1	0	0	−1	0	1	3	-
$z_j - c_j$			−105		−108		208	*	
−100	x_5	0	3/4	−1/4	0	1	0	3	12/3
0	x_4	0	1/4	1/4	1	0	−1	4	16
8	x_1	1	1/4	1/4	0	0	0	7	28
$z_j - c_j$			-78	27			100	*	
5	x_2	0	1	−1/3	0	4/3	0	4	
0	x_4	0	0	1/3	1	−1/3	−1	3	
8	x_1	1	0	1/3	0	−1/3	0	6	
$z_j - c_j$				1		104	100	68	

* Pokud jsou v bázi pomocné proměnné, nemá smysl počítat hodnotu účelové funkce:

Pomocné proměnné nemají ekonomickou interpretaci a prohibitivní sazby byly zvoleny (M = – 100)

Podíl β_1 / α_{14} nemá smysl, protože koeficient $\alpha_{14} = -1$ je záporný.

Krok 6: **Sestavení a interpretace vektoru řešení**

a) Ve výsledném vektoru řešení budou pouze čísla ze sloupce xi odpovídající bázovým proměnným ze sloupce Z. U všech ostatních proměnných bude 0 – proměnné, které nejsou v bázi, nejsou pro maximalizaci zisku výhodné. Výsledný vektor řešení má tedy tvar:

$$\begin{array}{cccccccc} X = (& 6; & 4; & 0; & 3; & 0; & 0 &) \\ & x1 & x2 & x3 & x4 & x5 & x6 & \end{array}$$

X1 = optimální počet přístrojů typu A je 6
X2 = optimální počet přístrojů typu B je 4
X4 = překročení počtu přístrojů A nad 3 ks je o 3

Je úloha degenerovaná? Není, protože žádná z bazických proměnných (pravý sloupec) není nulová.
Má úloha alternativní řešení? Nemá, protože žádná z nebazických proměnných nemá v řádku zj-cj hodnotu 0.

Krok 7: **Dopočítám výsledný maximální zisk**

Zisk = 68 * 100 000 Kč = 6.800.000 Kč

Domácí úkol☺

Opište si na dva samostatné papíry pouze zadání předchozího Příkladu 1 a Příkladu 2 a zkuste je oba znovu samostatně spočítat BEZ koukání na postup. Až vyjdou oba správně přesně podle výsledku, úkol je splněn.

Převod Primárního modelu na Duální model

Příklad 3: Převeďte níže uvedený primární model lineárního programování na duální model.

Primární model

$6x_1 + 2x_2 - 5x_3 + 2x_4 \leq 20$

$-3x_1 + 5x_2 - 4x_3 + 9x_4 \geq 30$

$2x_1 + 7x_2 - 6x_3 + 2x_4 = 40$

$Z_{MAX} = 17x_1 + 25x_2 + 11x_3 + 30x_4$

$x_1, x_2, x_3, x_4 \geq 0$

Řešení:

Krok 1: Čísla nalevo z **řádků** omezujících podmínek primárního modelu (**oranžová**, **zelená**, **modrá** barva) dám nalevo do **sloupců** omezujících podmínek duálního modelu.

Krok 2: Čísla napravo z omezujících podmínek primárního modelu (**fialová** barva) dám do účelové funkce duálního modelu

Krok 3: Čísla z účelové funkce primárního modelu (**červená** barva) dám napravo do omezujících podmínek duálního modelu

Krok 4: Změním Z_{MAX} na Z_{MIN}

Krok 5: Znaménka z omezujících podmínek primárního modelu **otočím** a dám je do podmínek nezápornosti duálního modelu. V případě = píšeme u duálního modelu s.l. (sin limites = bez omezení)

Krok 6: Znaménka z podmínek nezápornosti primárního modelu **neotáčím** a dám je do omezujících podmínek duálního modelu

Duální model

$6y_1 - 3y_2 + 2y_3 \geq 17$

$2y_1 + 5y_2 + 7y_3 \geq 25$

$-5y_1 - 4y_2 - 6y_3 \geq 11$

$2y_1 + 9y_2 + 2y_3 \geq 30$

$Z_{MIN} = 20y_1 + 30y_2 + 40y_3$

$y_1 \geq 0$, $y_2 \leq 0$, $y_3 =$ s.l. (sin limites)
bez omezení

Převod z Minimalizace na Maximalizaci

Krok 1-3 Stejně jako u převodu z Maximalizace na Minimalizaci

Krok 4: Změním Z_{MIN} na Z_{MAX}

Krok 5: Znaménka z omezujících podmínek primárního modelu **neotáčím** a dám je do podmínek nezápornosti duálního modelu

Krok 6: Znaménka z podmínek nezápornosti primárního modelu **otočím** a dám je do omezujících podmínek duálního modelu

5. Postoptimalizační analýza simplexové tabulky

Příklad 1: Krmivářská skupina chce na 10 ha orné půdy vyprodukovat co nejvíce krmných jednotek. Má k dispozici disponibilní pracovní hodiny: v květnu 140, v srpnu 150 a v září 130. Chce pěstovat 4 plodiny A, B, C, D.

Na 1ha	A	B	C	D
Květen	20	10	-	-
Srpen (v hod.)	10	10	10	10
Září	10	20	10	20
Krmné jednotky v t	10	8	2	1

Výchozí tabulka

c_B	B	10	8	2	1	0	0	0	0	b
		1	2	3	4	5	6	7	8	
0	5	1	1	1	1	1	0	0	0	10
0	6	20	10	0	0	0	1	0	0	140
0	7	10	10	10	10	0	0	1	0	150
0	8	10	20	10	20	0	0	0	1	130
$z_j - c_j$		−10	−8	−2	−1	0	0	0	0	0

Výsledná tabulka

c_B	B									x_B
0	5	0	0	2/3	2/3	1	−1/30	0	−1/30	1
10	1	1	0	−1/3	−1/3	0	1/15	0	−1/30	5
0	7	0	0	20/3	−10/3	0	−1/3	1	−1/3	60
8	2	0	1	2/3	2/3	0	−1/30	0	1/15	4
$z_j - c_j$		0	0	0	1	0	2/5	0	1/5	82

a) Vektor bazického (základního) řešení

Do vektoru bazického (základního) řešení dáme všechny hodnoty proměnných $x_1, \ldots, x_8$. U bázových proměnných bude hodnota z pravého sloupce, u nebázových bude hodnota 0.

X_B = (5;4;0;0;1;0;60;0)
X_1 = 5….plocha plodiny A
X_2 = 4….plocha plodiny B
X_5 = 1….nevyužitá disponibilní orná půda
X_7 = 60….nevyužité pracovní hodiny v srpnu

Hodnota účelové funkce je 82 (najdeme v pravém dolním rohu tabulky nebo spočítáme dosazením x z bazického řešení do účelové funkce).

b) Vektor obecného řešení

U bázových proměnných převedu všechny hodnoty proměnných z daného řádku na pravou stranu tabulky (kromě samotné proměnné, o kterou se jedná). U nebázových proměnných píšu jen jejich název.

$$X_O = \begin{pmatrix} 5 + \frac{1}{3}x3 + \frac{1}{3}x4 - \frac{1}{15}x6 + \frac{1}{30}x8 \\ 4 - \frac{2}{3}x3 - \frac{2}{3}x4 + \frac{1}{30}x6 - \frac{1}{15}x8 \\ x3 \\ x4 \\ 1 - \frac{2}{3}x3 - \frac{2}{3}x4 + \frac{1}{30}x6 + \frac{1}{30}x8 \\ x6 \\ 60 - \frac{20}{3}x3 + \frac{10}{3}x4 + \frac{1}{3}x6 + \frac{1}{3}x8 \\ x8 \end{pmatrix}$$

c) Matice báze B

Sloupce matice báze B najdeme ve výchozí simplexové tabulce nad sloupci jednotkových vektorů, které se nachází ve výsledné tabulce, tj. nad sloupci x5, x1, x7, x2. Pořadí je od sloupce s 1 v nejvyšším řádku až po 1 v nejnižším řádku.

$$\mathbf{B} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 20 & 0 & 10 \\ 0 & 10 & 1 & 10 \\ 0 & 10 & 0 & 20 \end{pmatrix}$$

d) Inverzní matice báze B^{-1}

Sloupce inverzní matice báze B^{-1} najdeme v těch sloupcích výsledné tabulky, v nichž ve výchozí tabulce byly jednotkové vektory, tj. ve sloupcích x5, x6, x7, x8. Pořadí je od sloupce s 1 v nejvyšším řádku až po 1 v nejnižším řádku.

$$\mathbf{B}^{-1} = \begin{pmatrix} 1 & -\frac{1}{30} & 0 & -\frac{1}{30} \\ 0 & \frac{1}{15} & 0 & -\frac{1}{30} \\ 0 & -\frac{1}{3} & 1 & -\frac{1}{3} \\ 0 & -\frac{1}{30} & 0 & \frac{1}{15} \end{pmatrix}$$

e) Duální hodnoty prvého a druhého druhu

Duální hodnoty čteme v řádku kritéria optimality zj-cj **výsledné** simplexové tabulky

- **Duální hodnoty prvého druhu** se nalézají ve sloupcích **strukturních** proměnných tj. (0;0;0;1) a udávají, o kolik se zhorší hodnota účelové funkce, zařadíme-li navíc jednu jednotku dané proměnné do řešení

Př: Každý hektar, na kterém bychom pěstovali nezařazenou plodinu D (x_4), by snížil vyrobené množství krmných jednotek o 1. Kdybychom tedy zařadili například nově (původně nevýhodnou) proměnnou D (x_4) o velikosti 1,5 ha, snížila by se účelová funkce o 1,5 krmných jednotek – tzv. suboptimální řešení viz dále. Kdybychom zařadili například nově (původně nevýhodnou) proměnnou C (x_3) o velikosti 1,5 ha, účelová funkce by se snížila o 0 krmných jednotek – tzv. alternativní řešení viz dále.

- **Duální hodnoty druhého druhu** se nalézají ve sloupcích **doplňkových** proměnných tj. (0; 2/5; 0; 1/5) a udávají, o kolik se zhorší hodnota účelové funkce, necháme-li jednu jednotku daného činitele nevyužitou.

Př: Každá pracovní hodina, o kterou přijdeme v měsíci květnu, přinese snížení krmných jednotek o 2/5. Každá pracovní hodina získaná navíc v měsíci září přinese zvýšení krmných jednotek o 1/5.

f) Analýza citlivosti optimálního řešení vzhledem k nezákladním (nebazickým) strukturním **proměnným x**

Jak se změní řešení, pokud zařadíme do řešení nějakou nezákladní (nebazickou, tedy původně nevýhodnou) strukturní proměnnou?

Nezákladní (nebazickou) strukturní proměnnou rozumíme takovou strukturní proměnnou, která předtím nebyla ve výsledné bázi (neukázala se jako výhodná). Pokud takovou proměnnou přece zařadíme do vektoru řešení, může být řešení buďto **suboptimální** (má hodnotu účelové funkce horší než vektor optimálního řešení) nebo **alternativní** (má hodnotu účelové funkce stejnou jako vektor optimálního řešení).

Postup pro výpočet nového řešení při zařazení nové proměnné: Od sloupce pravých stran výsledné simplexové tabulky odečteme hodnotu nově zařazované nebazické proměnné vynásobenou sloupcem výsledné tabulky pod zařazovanou nezákladní proměnnou

Př: Chceme z nějakých důvodů pěstovat (původně nevýhodnou) plodinu D na 1,5 = 3/2 ha. Tedy x_4 = 1,5 = 3/2 ha

$$\begin{array}{c} x_5\\ x_1\\ x_7\\ x_2 \end{array}\begin{pmatrix}1\\5\\60\\4\end{pmatrix} -3/2 \begin{pmatrix}2/3\\-1/3\\-10/3\\2/3\end{pmatrix} = \begin{pmatrix}0\\5{,}5\\65\\3\end{pmatrix}\begin{array}{c} x_5\\ x_1\\ x_7\\ x_2 \end{array}$$

Nové řešení má tvar:

$$X_B = \begin{array}{l}(5{,}5;\ 3;\ 0;\ 1{,}5;\ 0;\ 0;\ 65;\ 0)^T\\ \quad x_1\ \ x_2\ x_3\ \ x_4\ \ x_5\ x_6\ x_7\ x_8\end{array}$$

Dosazením do účelové funkce zjistíme, že její hodnota se **snížila na 80,5**. Řešení je **suboptimální**, získalo se totiž volbou nezákladní proměnné x_4, které odpovídá ve výsledné tabulce **nenulová** duální hodnota tj. $z_j - c_j = 1$.

Př: Chceme z nějakých důvodů pěstovat (původně nevýhodnou) plodinu C na 1,5 ha. Tedy x_3 = 1,5 = 3/2 ha

$$\begin{array}{c} x_5\\ x_1\\ x_7\\ x_2 \end{array}\begin{pmatrix}1\\5\\60\\4\end{pmatrix} -3/2 \begin{pmatrix}2/3\\-1/3\\20/3\\2/3\end{pmatrix} = \begin{pmatrix}0\\5{,}5\\50\\3\end{pmatrix}\begin{array}{c} x_5\\ x_1\\ x_7\\ x_2 \end{array}$$

Nové řešení má tvar:

$$X_B = \begin{array}{l}(5{,}5;\ 3;\ 1{,}5;\ 0;\ 0;\ 0;\ 50;\ 0)^T\\ \quad x_1\ \ x_2\ \ x_3\ \ x_4\ x_5\ x_6\ x_7\ x_8\end{array}$$

Dosazením do účelové funkce zjistíme, že její hodnota **zůstala 82.** Řešení je **alternativní**, získalo se totiž volbou nezákladní proměnné x_3, které odpovídá ve výsledné tabulce **nulová** duální hodnota tj. $z_j - c_j = 0$.

V jakém rozmezí mohou být jednotlivé nezákladní strukturní proměnné x_k, aby řešení zůstalo přípustné a ve stejné bázi (tzv. interval přípustných hodnot x_k)

Postup pro výpočet intervalu přípustných hodnot x_k:

Minimální přípustná hodnota zařazované nezákladní strukturní proměnné x_k je vždy 0.
Maximální přípustná hodnota zařazované nezákladní strukturní proměnné x_k se spočítá jako minimální hodnota z podílů pravých stran výsledné tabulky a pouze kladných koeficientů ve sloupci výsledné tabulky pod danou zařazovanou proměnnou.

Př: Vypočítejte interval přípustných hodnot pro zařazovanou proměnnou x_4

$$0 \le x_4 \le \min\left(\frac{1}{2/3}, \frac{4}{2/3}\right) = \frac{3}{2}$$

Tedy proměnnou x_4 můžeme zařadit pouze v rozmezí <0 ; 3/2>.

Př: Vypočítejte interval přípustných hodnot pro zařazovanou proměnnou x_3

$$0 \le x_3 \le \min\left(\frac{1}{2/3}, \frac{60}{20/3}, \frac{4}{2/3}\right) = \frac{3}{2}$$

Tedy proměnnou x_3 můžeme zařadit pouze v rozmezí <0 ; 3/2>.

g) Analýza citlivosti optimálního řešení vzhledem k **pravým stranám b** původní soustavy podmínek

Jak se změní řešení, pokud snížíme nebo zvýšíme hodnotu některé pravé strany b_i?

Postup pro výpočet nového řešení při změně pravé strany b_i: Od sloupce pravých stran výsledné simplexové tabulky odečteme (při snížení) nebo přičteme (při zvýšení) hodnotu změny pravé strany bi vynásobenou sloupcem výsledné tabulky pod doplňkovou proměnnou odpovídající dané pravé straně.

Př: Pokud se počet pracovních hodin v květnu sníží o 30 (140-30 = 110), nové bázové řešení x_B spočítáme následovně:

$$\begin{array}{c} x_5 \\ x_1 \\ x_7 \\ x_2 \end{array} \begin{pmatrix} 1 \\ 5 \\ 60 \\ 4 \end{pmatrix} - 30 . \begin{pmatrix} -1/30 \\ 1/15 \\ -1/3 \\ -1/30 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 70 \\ 5 \end{pmatrix} \begin{array}{c} x_5 \\ x_1 \\ x_7 \\ x_2 \end{array}$$

Nové řešení má tvar:

$$X_B = (\underset{x_1}{3};\ \underset{x_2}{5};\ \underset{x_3}{0};\ \underset{x_4}{0};\ \underset{x_5}{2};\ \underset{x_6}{0};\ \underset{x_7}{70};\ \underset{x_8}{0})$$

Dosazením do účelové funkce zjistíme, že její hodnota bude 70.

V jakém rozmezí mohou být jednotlivé **pravé strany**, aby řešení zůstalo ve stejné bázi (tzv. interval přípustných hodnot pro pravou stranu b_k)

Úkolem je nyní zjistit rozsah pravých stran tak, aby nedošlo ke změně báze. Změnu pravých stran si vyjádříme pomocí vektoru parametrů $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$. Např $\lambda_2 = 2$ znamená zvýšení pravé strany b_2 o 2 jednotky.

Postup pro výpočet intervalu přípustných hodnot pro parametry λ_k.

Minimální hodnotu spočítáme jako maximum ze záporných podílů pravých stran výsledné tabulky s kladnými čísly ve sloupci matice B^{-1}, který odpovídá pořadí dané pravé strany. Např. Dolní mez -75 pro druhou pravou stranu b_2 vznikla jako maximum ze záporných podílů čísel 1;5;60;4 s kladnými čísly ve druhém sloupci matice B^{-1},kde je však pouze jedno kladné číslo 1/15. Kdyby jich bylo více, bylo by to maximum z více čísel. Pokud nenajdu žádné kladné číslo, je dolní mez - ∞.

Maximální hodnotu spočítáme jako minimum ze záporných podílů pravých stran se zápornými čísly ve sloupci matice B^{-1}, který odpovídá pořadí dané pravé strany. Např. Horní mez 30 pro druhou pravou stranu b_2 vznikla jako minimum z podílů čísel 1;5;60;4 se zápornými čísly ve druhém sloupci matice B^{-1},kde jsou tři záporná čísla -1/30, -1/3, -1/30. Pokud nenajdu žádné záporné číslo, je horní mez ∞.

$$-1 \le \lambda_1 \langle \infty$$

$$-\frac{5}{1/15} = -75 \le \lambda_2 \le \min\left(\frac{1}{1/30}, \frac{60}{1/3}, \frac{4}{1/30}\right) = 30$$

$$-60 \le \lambda_3 \langle \infty$$

$$-\frac{4}{1/15} = -60 \le \lambda_4 \le \min\left(\frac{1}{1/30}, \frac{5}{1/30}, \frac{60}{1/3}\right) = 30$$

Tedy Např. Pravá strana b_2 může být v intervalu <65;170>

h) Analýza citlivosti optimálního řešení vzhledem ke změnám **koeficientů (cen) c_j** účelové funkce

V jakém rozmezí mohou být jednotlivé **ceny** strukturních proměnných, aby řešení zůstalo ve stejné bázi (tzv. interval přípustných hodnot pro cenu c_j)

Úkolem je zjistit, v jakém intervalu se mohou pohybovat ceny strukturních proměnných v účelové funkci, aby nedošlo ke změně báze (u nestrukturních nemá smysl uvažovat).
Změny cen si vyjádříme pomocí vektoru parametrů $v = (v_1, v_2, \ldots, v_p)$. Např $v_2 = 5$ znamená zvýšení ceny druhé proměnné v účelové funkci o 5 jednotek.

Výpočet intervalu přípustných hodnot pro jednotkovou změnu c_j nebazické strukturní proměnné x_j:

$-\infty \le v_j \le z_j - c_j$

Př: Pro nebazické strukturní proměnné x_3 a x_4 platí:

$-\infty \le v_3 \le 0$ tedy $c_3 \le 2+0 = 2$

$-\infty \le v_4 \le 1$ tedy $c_4 \le 1+1 = 2$

Výpočet intervalu přípustných hodnot pro jednotkovou změnu cj bazické strukturní proměnné xj

Minimální hodnotu spočítáme jako maximum ze záporných podílů řádku zj-cj s kladnými čísly v řádku výsledné tabulky, který odpovídá dané bazické strukturní proměnné.
Maximální hodnotu spočítáme jako minimum ze záporných podílů řádku zj-cj se zápornými čísly v řádku výsledné tabulky, který odpovídá dané bazické strukturní proměnné.

Poznámka: Hodnoty 0 v řádku zj-cj při podílech ignorujeme

Př: Pro bazické strukturní proměnné x_1 a x_2 platí:

$\max(-\frac{2/5}{1/15}) = -6 \le v_1 \le \min(-\frac{1}{-1/3}; -\frac{1/5}{-1/30}) = 3$ ……… c1 = 10, tedy $4 \le C_1 \le 13$

$\max(-\frac{1}{2/3};\ -\frac{1/5}{1/15}) = -3/2 \le v_2 \le \min(-\frac{2/5}{-1/30}) = 12$ ……… c2 = 8, tedy $6{,}5 \le C_2 \le 20$

6. Jednostupňová dopravní úloha (JDÚ) – výchozí řešení

Model Jednostupňové dopravní úlohy: Je dáno m dodavatelů D_1, D_2, .., D_m a n spotřebitelů S_1,S_2, …S_n. Dodavatelé mají kapacity zboží a_1, a_2, …a_m a spotřebitelé mají požadavky na zboží b_1, b_2, …b_n. Cena dopravy (vzdálenost) mezi dodavatelem D_i a spotřebitelem S_j je rovna c_{ij}. Cílem úlohy je minimalizovat přepravní náklady (celkový počet tunokilometrů).

Příklad 1:

Máme tabulku vzdáleností od dodavatelů ke spotřebitelům

MIN!	S_1	S_2	S_3	S_4	
D_1	20	15	15	11	100
D_2	17	12	12	19	250
D_3	23	13	8	17	140
	80	200	110	100	

Výchozí řešení - Vytvoříme pomocí některé z níže uvedených tří metod:

1) Metoda Severozápadního rohu

Krok 1: Zjistíme, zda je úloha vyvážená (vybilancovaná).

MIN!	S_1	S_2	S_3	S_4	
D_1	20	15	15	11	100
D_2	17	12	12	19	250
D_3	23	13	8	17	140
	80	200	110	100	490 / 490

Krok 2: Navezeme maximum do levého horního (severozápadního) rohu a dodavatele nebo spotřebitele, který se naplní, vyškrtneme.

MIN!	S_1	S_2	S_3	S_4	
D_1	80 ^{20}	20 ^{15}	^{15}	^{11}	100
D_2	^{17}	180 ^{12}	^{12}	^{19}	250
D_3	^{23}	^{13}	^{8}	^{17}	140
	80	200	110	100	

Krok 3: Tím se tabulka zmenší a pokračujeme dále až do rozvezení všeho zboží. Při kontrole nám musí množství rozvezeného zboží v tabulce odpovídat celkovému rozváženému množství v pravém dolním rohu tabulky.

Krok 4: Vypočítáme hodnotu účelové funkce (výsledný počet tunokilometrů)

Z = 80.20 + 20.15 + 180.12 + 70.12 + 40.8 + 100.17 = 6920 tunokilometrů

MIN!	S_1	S_2	S_3	S_4	
D_1	80 ^{20}	20 ^{15}	^{15}	^{11}	100
D_2	^{17}	180 ^{12}	70 ^{12}	^{19}	250
D_3	^{23}	^{13}	40 ^{8}	100 ^{17}	140
	80	200	110	100	

2) Indexová metoda

Krok 1: Zjistíme, zda je úloha vyvážená (vybilancovaná).

MIN!	S_1	S_2	S_3	S_4	
D_1	^{20}	^{15}	^{15}	^{11}	100
D_2	^{17}	^{12}	^{12}	^{19}	250
D_3	^{23}	^{13}	^{8}	^{17}	140
	80	200	110	100	490 / 490

Krok 2: Najdeme buňku s nejmenším indexem (vzdáleností) a obsadíme maximálním množstvím zboží. Nuly u případného fiktivního dodavatele či spotřebitele zpočátku ignorujeme a obsazujeme až nakonec. Dodavatele nebo spotřebitele, který se naplní, vyškrtneme.

MIN!	S_1	S_2	S_3	S_4	
D_1	(20)	(15)	(15)	(11)	100
D_2	(17)	(12)	(12)	(19)	250
D_3	(23)	(13)	110 (8)	(17)	140
	80	200	110	100	

Krok 3: Tím se tabulka zmenší a pokračujeme dále podle nejmenšího indexu až do rozvezení všeho zboží. Při kontrole nám musí množství rozvezeného zboží v tabulce odpovídat celkovému rozváženému množství v pravém dolním rohu tabulky.

MIN!	S_1	S_2	S_3	S_4	
D_1	(20)	(15)	(15)	100 (11)	100
D_2	50 (17)	200 (12)	(12)	(19)	250
D_3	30 (23)	(13)	110 (8)	(17)	140
	80	200	110	100	

Poznámka: Při tvorbě výchozího řešení došlo k tzv. **degeneraci řešení** = ve výsledné tabulce neodpovídá počet obsazených buněk počtu m + n – 1 = 3 + 4 – 1 = 6 (m = počet dodavatelů, n = počet spotřebitelů). Nastává, pokud po odsazení jedné buňky vyškrtnu najednou dodavatele i spotřebitele.

3) Metoda VAM (Vogelova metoda)

Krok 1: Zjistíme, zda je úloha vyvážená (vybilancovaná).

MIN!	S_1	S_2	S_3	S_4	
D_1	(20)	(15)	(15)	(11)	100
D_2	(17)	(12)	(12)	(19)	250
D_3	(23)	(13)	(8)	(17)	140
	40	200	120	130	490 / 490

Krok 2: Vypočteme rozdíly mezi dvěma nejnižšími čísly ve všech řádcích a sloupcích – získáme tak **řádkové a sloupcové diference.**

MIN	S_1	S_2	S_3	S_4		Δr
D_1	20	15	15	11 100	100	4
D_2	17	12	12	19	250	0
D_3	23	13	8	17	140	5
	40	200	120	130		
Δs	3	1	4	6		

Krok 3: V řádku nebo sloupci, kde je největší diference se najde nejmenší číslo a obsadí se maximálním množstvím zboží. Dodavatel nebo odběratel, který je naplněn se vyškrtne. Vznikne nová tabulka, ve které musíme opět přepočítat nové diference. Takto postupujeme stejným postupem dále až do rozvezení celé kapacity zboží. Pokud mi už zbude poslední řádek či sloupec a nejdou spočítat diference, obsadím podle nejmenší vzdálenosti.

MIN	S_1	S_2	S_3	S_4		Δr
D_1	20	15	15	11 100	100	4
D_2	17 40	12	12	19	250	0
D_3	23	13	8	17	140	5
	40	200	120	130		
Δs	~~3~~, 6	~~1~~, 1	~~4~~, 4	~~6~~, 2		

MIN	S_1	S_2	S_3	S_4		Δr
D_1	20	15	15	11 / 100	100	4
D_2	17 / 40	12	12	19	250	0
D_3	23	13	8 / 120	17	140	5
	40	200	120	130		
Δs	~~3~~,6	~~1~~,1	~~4~~,4	~~6~~,2		

MIN	S_1	S_2	S_3	S_4		Δr
D_1	20	15	15	11 / 100	100	4
D_2	17 / 40	12 / 200	12	19	250	~~0~~,7
D_3	23	13	8 / 120	17	140	~~5~~,4
	40	200	120	130		
Δs	~~3~~,6	~~1~~,1	~~4~~,4	~~6~~,2		

MIN	S_1	S_2	S_3	S_4		Δr
D_1	20	15	15	11 / 100	100	4
D_2	17 / 40	12 / 200	12	19 / 10	250	~~0~~,7
D_3	23	13	8 / 120	17 / 20	140	~~5~~,4
	40	200	120	130		
Δs	~~3~~,6	~~1~~,1	~~4~~,4	~~6~~,2		

Řešení, které jsme získali výše uvedenými třemi metodami je tzv. **výchozí řešení** (není ještě úplně nejlepší). To musíme dále ještě optimalizovat pomocí tzv. **Dantzigova testu optimality**

7. Jednostupňová dopravní úloha – optimalizace

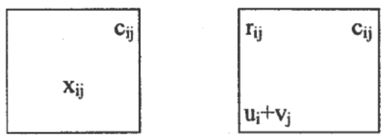
Dantzigův test optimality – Modifikovaná distribuční metoda (MODI)

MIN	S_1	S_2	S_3	S_4	S_5	
D_1	7 700	11 300	6	2	15	1 000
D_2	14	9 1 000	2 1 000	5	8	2 000
D_3	6	4	7 500	13 950	9 800	2 250
	700	1 300	1 500	950	800	

Krok 1: Po stranách tabulky dopočítáme postupně u a v. Položíme $u_1 = 0$ (pokud by se nepodařilo dopočítat, zvolíme jako 0 nějaké jiné u nebo v). Postupně dopočítáváme u a v, aby platilo u+v = c, ale pouze pro buňky, ve kterých je nějaké zboží.

Krok 2: V buňkách, kde není zboží, do **levého dolního rohu** napíšeme součty u+v = z. Do **levého horního rohu** napíšeme rozdíly r = z-c. Tedy: **levý horní roh = levý dolní - pravý horní roh**

a) Bazická proměnná b) Nebazická proměnná

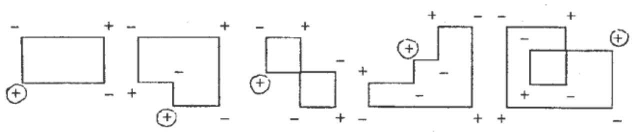


	S_1	S_2	S_3	S_4	S_5		u_i
D_1	7 700	11 300	-2 6 4	8 2 10	-9 15 6	1 000	volíme $u_i = 0$
D_2	-9 14 5	9 1 000	2 1 000	3 5 8	-4 8 4	2 000	-2
D_3	4 6 10	10 4 ⊕ 14	7 500	13 950	9 800	2 250	3
	700	1 300	1 500	950	800		
v_j	7	11	4	10	6		
	v_1	v_2	v_3	v_4	v_5		

Krok 3: Kdyby už vyšly všechny **levé horní rohy** r záporné nebo 0, je řešení optimální, nemusíme dál upravovat a máme výsledek. Pokud vyjdou nějaká kladná r, musíme pokračovat.

Krok 4: Vybereme buňku s největším kladným **levým horním** rohem r (10) a do této buňky dáme značku ⊕. Tam budeme přesouvat zboží pomocí Dantzigových uzavřených obvodů.

Pravidla: Můžeme postupovat pouze rovně (nahoru, dolů, doleva, doprava) ne šikmo a pouze po buňkách, kde je zboží. Následně propojíme trasu čarami a buňky střídavě označíme + a -.



Vybereme nejmenší množství v buňce s označením – (u nás to je 500) a přesuneme toto množství po buňkách Dantzigova obvodu. U + přičteme, u – odečteme.

	S_1	S_2	S_3	S_4	S_5		u_i
D_1	7 700	11 300	-2 6 4	8 2 10	-9 15 6	1 000	volíme $u_i = 0$
D_2	-9 14 5	9 1 000 -	2 1 000 +	3 5 8	-4 8 4	2 000	-2
D_3	4 6 10	10 4 ⊕ 14	7 - 500	13 950	9 800	2 250	3
	700	1 300	1 500	950	800		
v_j	7	11	4	10	6		
	v_1	v_2	v_3	v_4	v_5		

Krok 5: Vznikne nám nová tabulka (viz podrobné dopočítání na doučování), v ní pokračujeme stejným způsobem (znova u, v, dopočítám rohy buněk atd….). Provádíme tak dlouho, dokud nám nevyjdou všechny levé horní rohy r záporné. V tu chvíli jsme hotovi a máme optimální řešení.

Krok 6: Dopočítáme výslednou hodnotu účelové funkce

Rozbor optimálního řešení dopravní úlohy

Perspektivita spoje (Ir_{ij}I - absolutní hodnota levých horních rohů rij) - je to hodnota testu optimality v optimálním řešení u nerealizovaného spoje (tedy u prázdných buňek, kde nic nevezu). U optimálních spojů (obsazené buňky) je perspektivita rovná 0. Určuje míru zhoršení účelové funkce, pokud bychom využili tento spoj. Čím je toto číslo menší, tím je spoj výhodnější. Je li tato hodnota malá, můžeme po této trase převážet zboží se zanedbatelným zhoršením účelové funkce. Čím je tedy hodnota Ir_{ij}I nižší, tím je spoj perspektivnější. Je-li hodnota Ir_{ij}I dokonce nulová, existuje rovnocenné alternativní řešení. Po tomto spoji můžeme tedy převážet zboží až do výše propustnosti spoje (viz dále) a účelová funkce se nezmění.

Propustnost spoje – říká, jaké je maximální možné množství zboží, které může být na daném spoji přepraveno. U optimálních spojů (obsazené buňky) je propustnost rovná přímo přepravovanému množství zboží (uprostřed buňky). U nerealizovaných spojů (prázdné buňky) se propustnost rovná hodnotě, kterou je možno přesunout po Dantzigově obvodu sestrojeném k tomuto spoji (tedy minimální hodnotě v buňkách tohoto obvodu označených -).

Nevyvážené (nevybilancované) dopravní úlohy

Nevyvážená (nevybilancovaná) dopravní úloha = kapacity dodavatelů nejsou rovny kapacitám spotřebitelů. Řešíme přidáním fiktivního dodavatele nebo fiktivního spotřebitele, který má kapacitu rovnou rozdílu, který chybí do vyvážení.

MIN	S_1	S_2	S_3	S_4	S_5	
D_1	7	11	6	2	15	1 000
D_2	14	9	2	5	8	2 000
D_3	6	4	7	13	9	500
	1 000	1 000	2 000	450	800	5250 / 3 500

MIN	S_1	S_2	S_3	S_4	S_5	
D_1	7	11	6	2	15	1 000
D_2	14	9	2	5	8	2 000
D_3	6	4	7	13	9	500
D_4 (fiktivní)	0	0	0	0	0	1 750
	1 000	1 000	2 000	450	800	5250 / 5250

Degenerace řešení v dopravních úlohách

Degenerace řešení nastává, pokud je počet buněk obsazených zbožím v dopravní tabulce menší než m+n-1, kde m = počet řádků, n = počet sloupců. Neboli v bázi (řešení) je menší počet buněk, než by měl být.

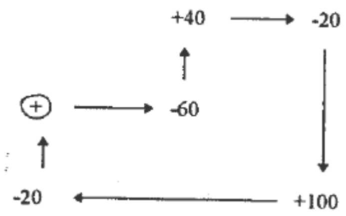
Degenerace řešení může vzniknout 2 způsoby:

1) Degenerace při konstrukci výchozího řešení – nastává, pokud obsadíme buňku, která najednou naplní dodavatele i spotřebitele se stejnými kapacitami. Vyškrtnu najednou oba dva (dodavatele i spotřebitele), ubyde mi jedno řešení z báze, a mám méně než m+n-1 plných buněk.

		S_j		a_i
D_i		700		700
		700		
		b_j		

Náprava: U dodavatele, nebo spotřebitele zvýšíme kapacitu o malý kousek ε (zanedbatelné množství např 1 g).

2) Degenerace při přesunu po Dantzigově obvodu – nastává, pokud v rozích označených zápornými znaménky jsou 2 stejné hodnoty. Při přesunu zboží po obvodu se namísto obvyklé 1 buňky vynulují 2 a počet obsazených buněk klesne pod m+n-1.



Náprava: K jednomu ze 2 stejných čísel přidáme malý kousek +ε (zanedbatelné množství např. 1 g).

Domácí úkol☺

Příklad: Jsou dány vzdálenosti mezi dodavateli a spotřebiteli:

	Prodejna A	Prodejna B	Prodejna C	ai
Výroba A	7	5	6	100
Výroba B	7	6	9	150
bj	100	80	120	

a) Je úloha vyvážená? Zdůvodněte
b) Pomocí VAM přejděte na výchozí řešení
c) Přejděte na optimální řešení a napište optimální hodnotu účelové funkce

8. Dvoustupňová dopravní úloha (DDÚ)

Model Dvoustupňové dopravní úlohy: Je dáno m dodavatelů D_1, D_2, .., D_m, kteří mají zboží o kapacitách a_1, a_2, …a_m. Toto zboží se má přepravit přes mezisklady M_1, M_2, .., M_n o kapacitách b_1, b_2, …b_n ke spotřebitelům S_1,S_2, …S_r, jejichž požadavky jsou p_1, p_2, …p_r. Cena dopravy (vzdálenost) mezi dodavatelem D_i a meziskladem M_j je rovna c_{ij}. Cena dopravy (vzdálenost) mezi meziskladem M_j a spotřebitelem S_k je rovna d_{jk}. Cílem úlohy je minimalizovat přepravní náklady (celkový počet tunokilometrů).

Příklad:
Na čtyřech menších nádražích Týniště, Bělohrádek, Letohrad, Mohelnice (dodavatelé) jsou odstavené staré lokomotivy, počty a vzdálenosti jsou v tabulce. Tyto lokomotivy mají zamířit do železničních muzeí v Praze, Jaroměři a Rosicích (spotřebitelé). Předtím je však třeba tyto lokomotivy zrenovovat v DKV v Nymburce, České Třebové a Olomouci (mezisklady). Navrhněte plán přesunů lokomotiv. Minimalizujte ujetou vzdálenost.

Zadání (vzdálenosti mezi městy a počty lokomotiv)							
	Nymburk	Česká Třebová	Olomouc	Praha	Jaroměř	Rosice	Lokomotiv
Týniště nad Orlicí	87	49	135				7
Borohrádek	95	41	127				5
Letohrad	129	24	110				8
Mohelnice	177	52	58				3
Nymburk				50	83	67	10
Česká Třebová				164	99	62	10
Olomouc				250	185	148	5
Počet lokomotiv	10	10	5	8	7	8	

a) O jaký typ úlohy se jedná? (5b)

b) Do prázdných míst v tabulce doplňte odpovídající ceny tras (5 b)

c) Máte možnost naplánovat kapacity renovačních pracovišť DKV v Nymburce, České Třebové a Olomouci (mezisklady) zcela libovolně, opět za účelem minimalizace ujeté vzdálenosti při přejezdech lokomotiv. Jak byste museli změnit hodnoty ve vstupní tabulce? O jaký typ úlohy se jedná? (10 b)

Řešení (viz doučování):

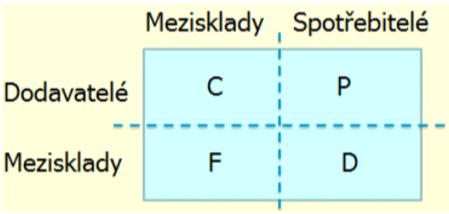
Krok 1: Jedná se o Dvoustupňovou dopravní úlohu převedenou z původního zadání například

		Mezisklady			
		M1	M2	M3	
	Dodavatelé	Nymburk	Česká Třebová	Olomouc	**Lokomotiv**
D1	Týniště n.O	87	49	135	**7**
D2	Borohrádek	95	41	127	**5**
D3	Letohrad	129	24	110	**8**
D4	Mohelnice	177	52	58	**3**
	Spotřebitelé				
S1	Praha	50	164	250	**8**
S2	Jaroměř	83	99	185	**7**
S3	Rosice	67	62	148	**8**
	Lokomotiv	**10**	**10**	**5**	

na Jednostupňovou dopravní úlohu

Zadání (vzdálenosti mezi městy a počty lokomotiv)							
	Nymburk	Česká Třebová	Olomouc	Praha	Jaroměř	Rosice	Lokomotiv
Týniště nad Orlicí	87	49	135				7
Borohrádek	95	41	127				5
Letohrad	129	24	110				8
Mohelnice	177	52	58				3
Nymburk				50	83	67	10
Česká Třebová				164	99	62	10
Olomouc				250	185	148	5
Počet lokomotiv	10	10	5	8	7	8	

pomocí schématu



Krok 2: Do prázdných buněk vpravo nahoře dáme prohibitivní sazby např. 1000 (od dodavatelů přímo ke spotřebitelům se u DDÚ nic neveze). Do prázdných buněk vlevo dole dáme prohibitivní sazby např. 1000 (z meziskladů do meziskladů se u DDÚ také neveze). Do diagonály dáme 0 (vzdálenost mezi meziskladem např. v České Třebové a České Třebové je 0).

Zadání (vzdálenosti mezi městy a počty lokomotiv)							
	Nymburk	Česká Třebová	Olomouc	Praha	Jaroměř	Rosice	Lokomotiv
Týniště nad Orlicí	87	49	135	1000	1000	1000	7
Borohrádek	95	41	127	1000	1000	1000	5
Letohrad	129	24	110	1000	1000	1000	8
Mohelnice	177	52	58	1000	1000	1000	3
Nymburk	0	1000	1000	50	83	67	10
Česká Třebová	1000	0	1000	164	99	62	10
Olomouc	1000	1000	0	250	185	148	5
Počet lokomotiv	10	10	5	8	7	8	

Krok 3: Převedli jsme Dvoustupňovou dopravní úlohu na Jednostupňovou dopravní úlohu. Řádky se nyní chovají jako dodavatelé v JDÚ, sloupce jako spotřebitelé v JDÚ. Dořešíme úlohu stejně jako vybilancovanou Jednostupňovou dopravní úlohu (výchozí řešení např. pomocí Severozápadní metody, optimální řešení pomocí Dantzigových uzavřených obvodů) - viz doučování. Pokud by úloha nebyla vybilancovaná, přidávaly by se fiktivní řádky či sloupce jako u JDÚ. Výsledné počty lokomotiv a výsledných cen po všech úpravách jsou uvedeny v tabulkách níže.

Počty lokomotiv, které se budou přesunovat mezi městy

	Nymburk	Česká Třebová	Olomouc	Praha	Jaroměř	Rosice	Lokomotiv
Týniště nad Orlicí	7	0	0				7
Borohrádek	3	2	0				5
Letohrad	0	8	0				8
Mohelnice	0	0	3				3
Nymburk	8	2	0	8	2	0	10
Česká Třebová	0	5	5	0	5	5	10
Olomouc	0	0	3	0	0	3	3
Počet lokomotiv	10	10	5	8	7	8	

Ujeté vzdálenosti a celková vzdálenost (cena kterou minimalizujeme)

	Nymburk	Česká Třebová	Olomouc	Praha	Jaroměř	Rosice	Lokomotiv	
Týniště nad Orlicí	609	0	0				7	
Borohrádek	285	82	0				5	
Letohrad	0	192	0				8	
Mohelnice	0	0	174				3	
Nymburk				400	166	0	10	
Česká Třebová				0	495	310	10	
Olomouc				0	0	444	5	Cena
Počet lokomotiv	10	10	5	8	7	8		3157

Krok 3: Pokud bychom chtěli nastavit optimálně kapacity meziskladů za účelem minimalizace vzdáleností (viz podotázka c v zadání), museli bychom nejdříve kapacity meziskladů v úloze předimenzovat, aby se do každého vešly případně úplně všechny lokomotivy (např. dáme kapacity všech meziskladů 100) a vyřešíme úlohu znova. Od předimenzovaných kapacit meziskladů 100 finálně vždy odečteme nově vzniklé hodnoty na diagonále v šedých polích, kde na počátku byly nuly. Pro zájemce viz video „DDÚ excel" v souborech na Fb skupině EMM I. Jedná se o úlohu dimenzování meziskladů ve dvoustupňové dopravní úloze převedené na jednostupňovou dopravní úlohu.

Zadání (vzdálenosti mezi městy a počty lokomotiv)

	Nymburk	Česká Třebová	Olomouc	Praha	Jaroměř	Rosice	Lokomotiv
Týniště nad Orlicí	87	49	135				7
Borohrádek	95	41	127				5
Letohrad	129	24	110				8
Mohelnice	177	52	58				3
Nymburk				50	83	67	100
Česká Třebová				164	99	62	100
Olomouc				250	185	148	100
Počet lokomotiv	100	100	100	8	7	8	

9. Další dopravní úlohy

9.1 Přiřazovací úloha (Maďarská metoda)

Používá se k optimálnímu přiřazení n prvků k n místům. Je to v podstatě analogie dopravního problému, ale mám stejný počet dodavatelů a spotřebitelů a všichni mají kapacitu 1. Nejčastějším úkolem je rozvést při minimálních nákladech n strojů do n míst, přičemž máme v zadání jednotlivé vzdálenosti.

Důležitým termínem je tzv. **Nezávislá nula** – je to taková nula, která je sama v příslušném řádku nebo sloupci

Příklad 1: Máme tabulku vzdáleností mezi středisky a hony. Z každého střediska má být dopraven jeden traktor na jeden hon tak, aby počet ujetých km byl minimální.

Střediska	Hony				
	1.	2.	3.	4.	5.
1.	12	8	5	5	9
2.	5	3	11	15	18
3.	7	9	20	19	13
4.	11	17	14	8	8
5.	16	13	9	6	11

Krok 1: Provedeme řádkovou redukci tak, že od sazeb v jednotlivých řádcích odečteme vždy nejmenší sazbu (5,3,7,8,6) a dostaneme:

$$\begin{pmatrix} 7 & 3 & 0 & 0 & 4 \\ 2 & 0 & 8 & 12 & 15 \\ 0 & 2 & 13 & 12 & 6 \\ 3 & 9 & 6 & 0 & 0 \\ 10 & 7 & 3 & 0 & 5 \end{pmatrix}$$

V každém řádku i sloupci je již nula, nemusíme tedy už následně provádět sloupcovou redukci.

Krok 2: V redukované matici sazeb vybíráme nezávislé nuly (nuly, které jsou samy v řádku nebo sloupci). Nezávislou nulu vždy označíme rámečkem a škrtneme řadu (řádek či sloupec), která je **kolmá** na řadu ve které je nula sama (tedy tu druhou, kde jsou většinou ještě jiné nuly). Pokud je nula sama v řádku i sloupci (tzv. silně nezávislá nula), můžeme si vybrat, zda škrtneme řádek či sloupec (např. vždy řádek). Škrtnuté nuly (řady) už dále ignorujeme. Takto pokračujeme, dokud je možno vybírat nezávislé nuly. Postupujeme například vždy shora dolů po řádcích, abychom nic nepřeskočili, tedy v pořadí:

1: 1.ř 3.sl
2: 2.ř 2.sl
3: 3.ř 1.sl
4: 4.ř 5.sl
5: 5.ř 4.sl

$$\begin{pmatrix} 7 & 3 & 0 & 0 & 4 \\ 2 & 0 & 8 & 12 & 15 \\ 0 & 2 & 13 & 12 & 6 \\ 3 & 9 & 6 & 0 & 0 \\ 10 & 7 & 3 & 0 & 5 \end{pmatrix} \quad \begin{pmatrix} 7 & 3 & \boxed{0} & 0 & 4 \\ 2 & \boxed{0} & 8 & 12 & 15 \\ \boxed{0} & 2 & 13 & 12 & 6 \\ 3 & 9 & 6 & 0 & \boxed{0} \\ 10 & 7 & 3 & \boxed{0} & 5 \end{pmatrix}$$

Krok 3: Nezávislých nul je právě 5, získali jsme tedy optimální řešení. Vrátíme se k výchozí tabulce a určíme, jaké budou přesuny traktorů

Střediska	Hony				
	1.	2.	3.	4.	5.
1.	12	8	5 1	5	9
2.	5	3 1	11	15	18
3.	7 1	9	20	19	13
4.	11	17	14	8	8 1
5.	16	13	9	6 1	11

Optimální program přesunů traktorů tedy je:

z 1.střediska na 3. hon
z 2.střediska na 2. hon
z 3.střediska na 1. hon
z 4.střediska na 5. hon
z 5.střediska na 4. hon

Hodnota účelové funkce je Z = 5.1+3.1+7.1+8.1+6.1 = 29. Celkový minimální počet ujetých km je tedy 29.

Příklad 2: Z 6 závodů máme přesunout do 6 středisek kombajny, aby náklady na dopravu byly minimální. Náklady na dopravu mezi daným závodem a střediskem jsou uvedeny v tabulce.

	Střediska					
Závody	1.	2.	3.	4.	5.	6.
1.	21	15	10	9	14	18
2.	10	5	12	8	4	16
3.	11	9	15	17	20	25
4.	6	13	19	18	22	17
5.	10	12	14	20	21	15
6.	8	9	11	21	18	16

Krok 1: Provedeme řádkovou redukci tak, že od sazeb v jednotlivých řádcích odečteme vždy nejmenší sazbu (9,4,9,6,10,8) a dostaneme redukovanou matici:

$$\begin{pmatrix} 12 & 6 & 1 & 0 & 5 & 9 \\ 6 & 1 & 8 & 4 & 0 & 12 \\ 2 & 0 & 6 & 8 & 11 & 16 \\ 0 & 7 & 13 & 12 & 16 & 11 \\ 0 & 2 & 4 & 10 & 11 & 5 \\ 0 & 1 & 3 & 13 & 10 & 8 \end{pmatrix}$$

Krok 2: Protože 3. a 6. sloupec neobsahuje nulu, provedeme sloupcovou redukci tak, že v jednotlivých sloupcích odečteme nejnižší sazbu (0,0,1,0,0,5) a dostaneme:

$$\begin{pmatrix} 12 & 6 & 0 & 0 & 5 & 4 \\ 6 & 1 & 7 & 4 & 0 & 7 \\ 2 & 0 & 5 & 8 & 11 & 11 \\ 0 & 7 & 12 & 12 & 16 & 6 \\ 0 & 2 & 3 & 10 & 11 & 0 \\ 0 & 1 & 2 & 13 & 10 & 3 \end{pmatrix}$$

Krok 3: Vybíráme nezávislé nuly v tomto pořadí:

1: 1.ř 3.sl
2: 2.ř 5.sl
3: 3.ř 2.sl
4: 4.ř 1.sl
5: 5.ř 6.sl

$$\begin{pmatrix} 12 & 6 & 0 & 0 & 5 & 4 \\ 6 & 1 & 7 & 4 & 0 & 7 \\ 2 & 0 & 5 & 8 & 11 & 11 \\ 0 & 7 & 12 & 12 & 16 & 6 \\ 0 & 2 & 3 & 10 & 11 & 0 \\ 0 & 1 & 2 & 13 & 10 & 3 \end{pmatrix} \begin{pmatrix} 12 & 6 & \boxed{0} & 0 & 5 & 4 \\ 6 & 1 & 7 & 4 & \boxed{0} & 7 \\ 2 & \boxed{0} & 5 & 8 & 11 & 11 \\ \boxed{0} & 7 & 12 & 12 & 16 & 6 \\ 0 & 2 & 3 & 10 & 11 & \boxed{0} \\ 0 & 1 & 2 & 13 & 10 & 3 \end{pmatrix}$$

Krok 4: Vybrali jsme pouze 5 nezávislých nul, je potřeba jich vybrat 6. Provedeme **sekundární redukci** matice:

a) určíme nejmenší prvek z nepřeškrtnutých prvků (1)
b) prvky přeškrtnuté 1 x necháme beze změny
c) prvky přeškrtnuté dvakrát o minimální prvek zvětšíme
d) prvky nepřeškrtnuté o minimální prvek snížíme

$$\begin{pmatrix} 13 & 6 & 0 & 0 & 5 & 4 \\ 7 & 1 & 7 & 4 & 0 & 7 \\ 3 & 0 & 5 & 8 & 11 & 11 \\ 0 & 6 & 11 & 11 & 15 & 5 \\ 1 & 2 & 3 & 10 & 11 & 0 \\ 0 & 0 & 1 & 12 & 9 & 2 \end{pmatrix}$$

Krok 5: Vybíráme opět nezávislé nuly v tomto pořadí:

1: 1.ř 3.sl
2: 2.ř 5.sl
3: 3.ř 2.sl
4: 4.ř 1.sl
5: 5.ř 6.sl
a provedeme opět kontrolu pomocí krycích čar, postup byl správný.

$$\begin{pmatrix} 13 & 6 & 0 & 0 & 5 & 4 \\ 7 & 1 & 7 & 4 & 0 & 7 \\ 3 & 0 & 5 & 8 & 11 & 11 \\ 0 & 6 & 11 & 11 & 15 & 5 \\ 1 & 2 & 3 & 10 & 11 & 0 \\ 0 & 0 & 1 & 12 & 9 & 2 \end{pmatrix} \quad \begin{pmatrix} 13 & 6 & \boxed{0} & 0 & 5 & 4 \\ 7 & 1 & 7 & 4 & \boxed{0} & 7 \\ 3 & \boxed{0} & 5 & 8 & 11 & 11 \\ \boxed{0} & 6 & 11 & 11 & 15 & 5 \\ 1 & 2 & 3 & 10 & 11 & \boxed{0} \\ 0 & 0 & 1 & 12 & 9 & 2 \end{pmatrix}$$

Krok 7: Provedeme další sekundární redukci matice – nejmenší prvek z nepřeškrtnutých prvků je opět 1.

$$\begin{pmatrix} 14 & 7 & 0 & 0 & 5 & 4 \\ 8 & 2 & 7 & 4 & 0 & 7 \\ 3 & 0 & 4 & 7 & 10 & 10 \\ 0 & 6 & 10 & 10 & 14 & 4 \\ 2 & 3 & 3 & 10 & 11 & 0 \\ 0 & 0 & 0 & 11 & 8 & 1 \end{pmatrix}$$

Krok 8: Pokračujeme výběrem nezávislých nul v tomto pořadí:

1: 1.ř 4.sl
2: 2.ř 5.sl
3: 3.ř 2.sl
4: 4.ř 1.sl
5: 5.ř 6.sl
6: 6.ř 3.sl

$$\begin{pmatrix} 14 & 7 & 0 & 0 & 5 & 4 \\ 8 & 2 & 7 & 4 & 0 & 7 \\ 3 & 0 & 4 & 7 & 10 & 10 \\ 0 & 6 & 10 & 10 & 14 & 4 \\ 2 & 3 & 3 & 10 & 11 & 0 \\ 0 & 0 & 0 & 11 & 8 & 1 \end{pmatrix} \quad \begin{pmatrix} 14 & 7 & 0 & \boxed{0} & 5 & 4 \\ 8 & 2 & 7 & 4 & \boxed{0} & 7 \\ 3 & \boxed{0} & 4 & 7 & 10 & 10 \\ \boxed{0} & 6 & 10 & 10 & 14 & 4 \\ 2 & 3 & 3 & 10 & 11 & \boxed{0} \\ 0 & 0 & \boxed{0} & 11 & 8 & 1 \end{pmatrix}$$

Krok 9: Nezávislých nul je nyní 6, dostali jsme tedy optimální řešení. Výsledný plán přepravy je uveden v tabulce.

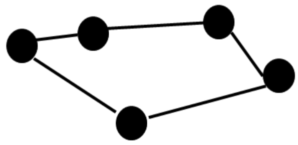
Závody	Střediska					
	1.	2.	3.	4.	5.	6.
1.	21	15	10	9 1	14	18
2.	10	5	12	8	4 1	16
3.	11	9 1	15	17	20	25
4.	6 1	13	19	18	22	17
5.	10	12	14	20	21	15 1
6.	8	9	11 1	21	18	16

Optimální program přesunů kombajnů tedy je:

z 1.závodu do 4. střediska
z 2.závodu do 5. střediska
z 3.závodu do 2. střediska
z 4.závodu do 1. střediska
z 5.závodu do 6. střediska
z 6.závodu do 3. střediska
Hodnota účelové funkce je Z = 9+4+9+6+15+11 = 54. Celkové minimální náklady jsou tedy 54 tisíc Kč.

9.2 Jednookruhový okružní problém - Metoda nejbližšího souseda

Používá se k nalezení nejkratší cesty, která obsahuje všechny vrcholy. Nejčastějším úkolem je projet co nejkratší cestou všechna města a vrátit se zpět do původního. Někdy též nazýváno **„Úloha obchodního cestujícího“.**



Příklad:
Turista vyjíždí z Catanzara a chce postupně navštívit všechna města v tabulce a vrátit se zpátky. Naplánujte trasu tak, aby ujel co nejméně kilometrů. Vzájemné vzdálenosti mezi městy jsou v tabulce

	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea
Catanzaro	-	97	76	158	152	94
Cosenza	97	-	116	187	95	124
Crotone	76	116	-	221	202	157
Reggio	158	187	221	-	242	104
Scalea	152	95	202	242	-	178
Tropea	94	124	157	104	178	-

Řešení:

Krok 1 : Procházím postupně všechna města. Ke každému najdu nejbližšího souseda, k němu opět nejbližšího souseda atd. až projedu všechna města a vrátím se zpět.

1) Catanzano-76-Crotone-116-Cosenza-95-Scalea-178-Tropea-104-Reggio-158-zpět: Celkem:**727 km**
2) Cosenza-95-Scalea-152-Catanzaro-76-Crotone-157-Tropea-104-Reggio-187-zpět: Celkem:771 km
3) Crotone-76-Catanzaro-94-Tropea-104-Reggio-187-Cosenza-95-Scalea-202-zpět: Celkem:758 km
4) Reggio-104-Tropea-94-Catanzaro-76-Crotone-116-Cosenza-95-Scalea-242 –zpět: Celkem:**727 km**
5) Scalea-95-Cosenza-97-Catanzaro-76-Crotone-157-Tropea-104-Reggio-242 –zpět: Celkem:771 km
6) Tropea-94-Catanzaro-76-Crotone-116-Cosenza-95-Scalea-242 - Reggio-104 –zpět: Celkem:**727 km**

Krok 2 : Vyberu nejkratší trasu (vyšly 3 trasy) a určím její (jejich) délku. Např. vybereme do výsledku trasu:
1) Catanzano-76-Crotone-116-Cosenza-95-Scalea-178-Tropea-104-Reggio-158-zpět: Celkem:**727 km**
nebo např. trasu
4) Reggio-104-Tropea-94-Catanzaro-76-Crotone-116-Cosenza-95-Scalea-242 –zpět: Celkem:**727 km**
u které ale musím ve výsledném řešení posunout začátek do Catanzara. Tedy:
Catanzaro-76-Crotone-116-Cosenza-95-Scalea-242 – Reggio-104-Tropea-94-zpět: Celkem:**727 km**

9.3 Jednookruhový okružní problém - Metoda VAM

Příklad: Turista vyjíždí z Catanzara a chce postupně navštívit všechna města v tabulce a vrátit se zpátky. Naplánujte trasu, aby ujel co nejméně kilometrů. Vzájemné vzdálenosti jsou v tabulce

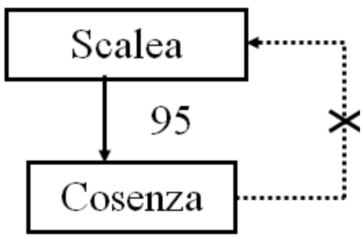
	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea
Catanzaro	-	97	76	158	152	94
Cosenza	97	-	116	187	95	124
Crotone	76	116	-	221	202	157
Reggio	158	187	221	-	242	104
Scalea	152	95	202	242	-	178
Tropea	94	124	157	104	178	-

Krok 1: Vypočítáme si řádkové a sloupcové diference (rozdíl dvou nejmenších vzdáleností v daném řádku či sloupci – viz VAM metoda u dopravní úlohy)

	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea	Řádkové diference
Catanzaro	-	97	76	158	152	94	18
Cosenza	97	-	116	187	95	124	2
Crotone	76	116	-	221	202	157	40
Reggio	158	187	221	-	242	104	54
Scalea	152	95	202	242	-	178	57
Tropea	94	124	157	104	178	-	10
Sloupcové diference	18	2	40	54	57	10	

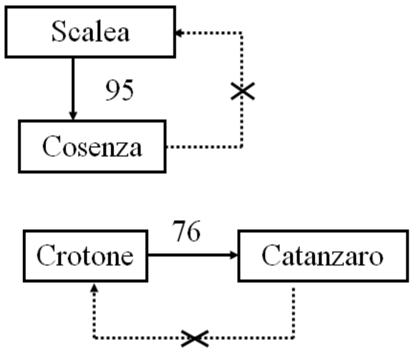
Krok 2: Vybereme nejmenší sazbu v řadě s maximální diferencí – tuto trasu vybereme do okruhu (trasa Scalea – Cosenza). Škrtáme řádek a sloupec u vybrané trasy a trasu, která předčasně uzavírá okruh (u izolovaných tras je to pouze cesta zpátky: Cosenza - Scalea).

	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea	
Catanzaro	-	97	76	158	152	94	18
Cosenza	97	-	116	187	95	124	2
Crotone	76	116	-	221	202	157	40
Reggio	158	187	221	-	242	104	54
Scalea	152	95	202	242	-	178	57
Tropea	94	124	157	104	178	-	10
	18	2	40	54	57	10	



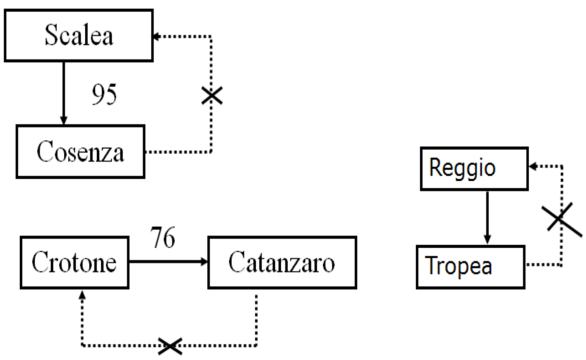
Krok 3: Přepočítáme opět diference již bez vyškrtnutého řádku a sloupce a volíme další trasu (Crotone – Catanzaro). Škrtáme opět řádek a sloupec u vybrané trasy a zpětnou trasu, která předčasně uzavírá okruh (Catanzaro – Crotone).

	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea	
Catanzaro	-	97	76	158	152	94	18,18
Cosenza	97	-	116	187	95	124	2,19
Crotone	76	116	-	221	202	157	40,81
Reggio	158	187	221	-	242	104	54,54
Scalea	152	95	202	242	-	178	57
Tropea	94	124	157	104	178	-	10,10
	18,18	2	40,40	54,54	57,26	10,10	



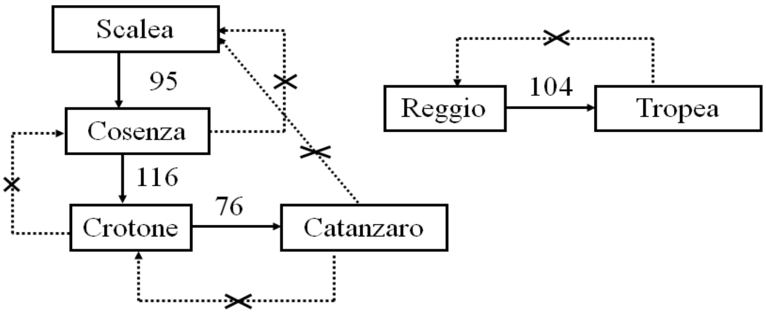
Krok 4: Přepočítáme opět diference a volíme další trasu (Reggio-Tropea). Škrtáme opět řádek a sloupec u vybrané trasy a zpětnou trasu, která předčasně uzavírá okruh (Tropea - Reggio).

	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea	
Catanzaro	-	97	76	158	152	94	18,18,58
Cosenza	97	-	116	187	95	124	2,19,8
Crotone	76	116	-	221	202	157	40,81
Reggio	158	187	221	-	242	104	54,54,117
Scalea	152	95	202	242	-	178	57
Tropea	94	124	157	104	178	-	10,10,53
	18,18	2	40,40 105	54,54, 54	57,26 26	10,10, 10	



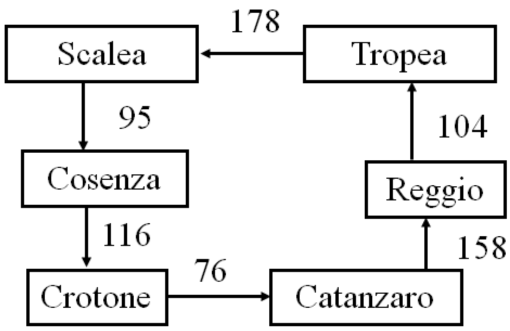
Krok 5: Přepočítáme opět diference a volíme další trasu (Cosenza - Crotone). Škrtáme opět řádek a sloupec u vybrané trasy a zpětnou trasu, která předčasně uzavírá okruh (Crotone - Cosenza) + trasu Catanzaro – Scalea, která by předčasně uzavřela okruh ještě před propojením všech míst.

	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea	
Catanzaro	-	97	76	158	152	94	18,18,58,6
Cosenza	97	-	116	187	95	124	2,19,8,71
Crotone	76	116	-	221	202	157	40,81
Reggio	158	187	221	-	242	104	54,54,117
Scalea	152	95	202	242	-	178	57
Tropea	94	124	157	104	178	-	10,10,53,21
	18,18	2	40,40, 105,41	54,54 54,29	57,26 26,26	10,10 10	



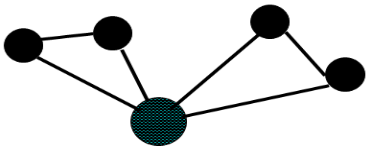
	Catanzaro	Cosenza	Crotone	Reggio	Scalea	Tropea
Catanzaro	-	97	76	158	152	94
Cosenza	97	-	116	187	95	124
Crotone	76	116	-	221	202	157
Reggio	158	187	221	-	242	104
Scalea	152	95	202	242	-	178
Tropea	94	124	157	104	178	-

Krok 6: Zbývají nám poslední dvě trasy, uzavřeme tedy okruh.



Krok 7: Navrhneme doporučenou trasu a vypočteme délku cesty:
Catanzano-158-Reggio-104 -Tropea-178-Scalea- 95-Cosenza-116-Crotone-76 -zpět: Celkem: **727 km**
Řešení vyšlo stejně jako v případě Metody nejbližšího souseda (akorát v obráceném směru).

9.4 Víceokruhový okružní problém (Mayerova metoda)



Nejčastější příčinou, proč je třeba někdy rozdělit jednookruhový okružní problém na více okruhů, je kapacitní omezení vozidel. Je tedy třeba naplánovat více okruhů (každý pro jedno vozidlo) tak, aby začínal a končil v centrálním místě. Suma kapacit (požadavků) všech necentrálních míst na jednom okruhu přitom nesmí překročit předem danou kapacitu vozidla

Příklad 1:

Firma rozváží zboží z centrálního skladu v Praze do skladů v krajských městech. K dispozici má vozidla o maximální kapacitě 30 t. Vzdálenosti jednotlivých měst a jejich požadavky jsou uvedeny níže:

	Praha	Č.Bud	Plzeň	Ústí	Hr.Kr.	Brno	Ostr.
Praha	-	140	94	92	112	202	362
Č. Budějovice	140	-	133	232	217	186	346
Plzeň	94	133	-	146	206	296	456
Ústí nad Labem	92	232	146	-	166	294	454
Hradec Králové	112	217	206	166	-	142	240
Brno	202	186	296	294	142	-	165
Ostrava	362	346	456	454	240	165	-

Město	Č. Bud	Plzeň	Ústí	Hr. Kr	Brno	Ostr
Požadavek	10	7	7	4	13	14

Řešení:

Krok 1: V tabulce víceokruhové úlohy si seřadíme místa (v řádcích i sloupcích) sestupně podle vzdálenosti od centrálního místa (to ale v tabulce vynecháme). Přidáme napravo sloupec požadavků.

Krok 2: Označíme první sloupec této tabulky (Ostravu - protože je nejdále od centrálního místa Prahy). Vyškrtneme první řádek a označíme tučně příslušný požadavek (14).

Krok 3: Pro každé z ostatních míst sečteme jeho požadavek s označeným (14) a u všech míst, kde tento součet bude větší než kapacita vozidla 30 t, vyškrtneme ve vybraném prvním sloupci buňku v příslušném řádku. V případě prvního sloupce většinou žádná buňka.

	Ostr.	Brno	Č.Bud.	Hr.Kr.	Plzeň	Ústí	Požadavek
Ostrava	-	~~165~~	~~346~~	~~240~~	~~456~~	~~454~~	**14**
Brno	165	-	186	142	296	294	13
Č. Budějovice	346	186	-	217	133	232	10
Hradec Králové	240	142	217	-	206	166	4
Plzeň	456	296	133	206	-	146	7
Ústí nad Labem	454	294	232	166	146	-	7

Krok 4: Z nevyškrtnutých prvků v prvním sloupci vybereme minimální prvek (v případě shody ten vyšší v tabulce). Ten označuje místo, které půjde jako další do okružní trasy (Brno). Odpovídající sloupec opět zvýrazníme, příslušný řádek vyškrtneme a zvýrazníme příslušný požadavek (13).

Krok 5: Sečteme všechny zvýrazněné požadavky (14+13) a pro ta místa, kde přičtením jejich požadavku s uvedeným součtem bude překročena kapacita 30 t, buňky v daných řádcích vyznačeného sloupce škrtneme. Z nevyškrtaných prvků v označeném sloupci bychom opět vybrali nejmenší prvek a zařadili do okružní trasy. Vidíme ale, že ve druhém sloupci jsou všechny prvky vyškrtnuty – máme první okruh: Praha, Ostrava, Brno.

	Ostr.	Brno	Č.Bud.	Hr.Kr.	Plzeň	Ústí	Požadavek
Ostrava	-	165	~~346~~	~~240~~	~~456~~	~~454~~	**14**
Brno	165	-	~~186~~	~~142~~	~~296~~	~~294~~	**13**
Č. Budějovice	346	~~186~~	-	217	133	232	10
Hradec Králové	240	~~142~~	217	-	206	166	4
Plzeň	456	~~296~~	133	206	-	146	7
Ústí nad Labem	454	~~294~~	232	166	146	-	7

Krok 6: Ve zbylé části tabulky hledáme opět stejným způsobem místa do dalších okružních tras.

	Ostr.	Brno	Č.Bud.	Hr.Kr.	Plzeň	Ústí	Požadavek
Ostrava	-	165	346	~~240~~	~~456~~	~~454~~	~~14~~
Brno	165	-	~~186~~	~~142~~	~~296~~	~~294~~	~~13~~
Č. Budějovice	346	~~186~~	-	~~217~~	~~133~~	~~232~~	**10**
Hradec Králové	240	~~142~~	217	-	206	166	4
Plzeň	456	~~296~~	133	206	-	146	7
Ústí nad Labem	454	~~294~~	232	166	146	-	7

	Ostr.	Brno	Č.Bud.	Hr.Kr.	Plzeň	Ústí	Požadavek
Ostrava	-	165	346	~~240~~	456	~~454~~	~~14~~
Brno	165	-	186	~~142~~	296	~~294~~	~~13~~
Č. Budějovice	346	186	-	~~217~~	133	~~232~~	10
Hradec Králové	240	142	217	-	206	166	4
Plzeň	456	296	133	~~206~~	-	~~146~~	7
Ústí nad Labem	454	294	232	166	146	-	7

	Ostr.	Brno	Č.Bud.	Hr.Kr.	Plzeň	Ústí	Požadavek
Ostrava	-	165	346	~~240~~	456	454	~~14~~
Brno	165	-	186	~~142~~	296	294	~~13~~
Č. Budějovice	346	186	-	~~217~~	133	232	10
Hradec Králové	240	142	217	-	206	166	4
Plzeň	456	296	133	~~206~~	-	146	7
Ústí nad Labem	454	294	232	~~166~~	146	-	7

Krok 7: Vyšel druhý (a poslední) okruh: Praha, Č. Bud, Plzeň, Ústí, Hr. Kr

Poznámka: Výše uvedený výpočet pro druhý okruh byl ukázán pouze pro pochopení metody. V tomto případě už však nebyl nutný, protože vzhledem ke kapacitě automobilu 30 tun je možno objet zbytek měst z Prahy už jen jedním okruhem a nemusíme nic počítat. Okruh Praha, Č. Bud, Plzeň, Ústí, Hr. Kr totiž nepřekročí kapacitu 30 a je tedy možné jej zajistit jen jedním autem.

Krok 8: V každém ze samostatných okruhů najdu optimální trasu jako u jednookružního dopravního problému (metodou nejbližšího souseda, nebo VAM metodou)

Výsledek:

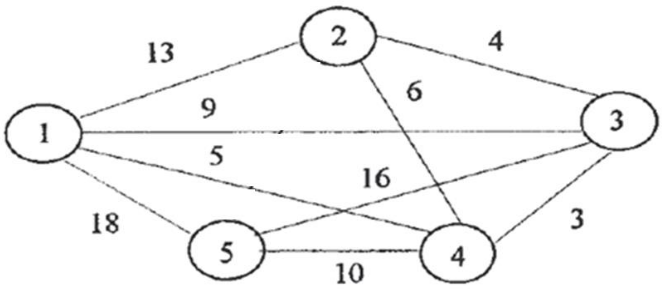
1. okruh: **Praha**-202-Brno-165-Ostrava-362 -zpět : **729 Km**
2. okruh: **Praha**-92-Ústí-146-Plzeň-133–Č. Bud–217-Hr. Kr.-112-zpět : **700 Km**

10. Teorie grafů

10.1 Minimální kostra (Kruskallův algoritmus)

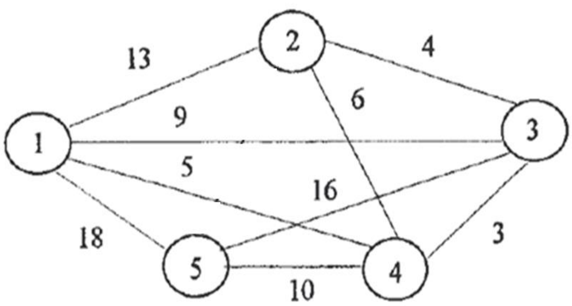
Používá se k nalezení nejkratšího propojení míst např. drahým kabelem. Na rozdíl od předchozího **okružního problému** se nevracíme zpátky a stačí jen místa propojit (**minimální kostra v grafu)**.

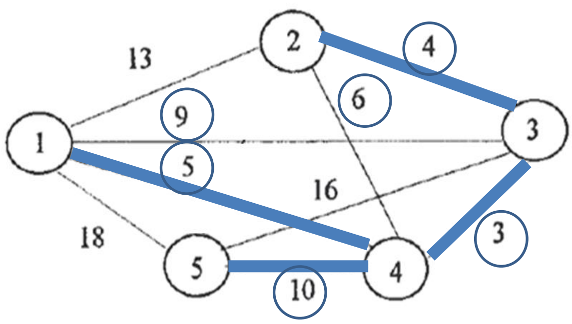
Příklad 1: Máme kabelem propojit 5 vesnic (svítidel na pozemku atd.), aby celková délka kabelu byla co nejkratší.



Řešení:

Krok 1: Procházíme v grafu hrany od nejkratší po nejdelší a zařazujeme (zvýrazňujeme) postupně jen ty, které nevytvoří okruh. Délky hran, které už jsme zařadili nebo přeskočili z důvodu hrozícího okruhu, dáváme do kroužku.





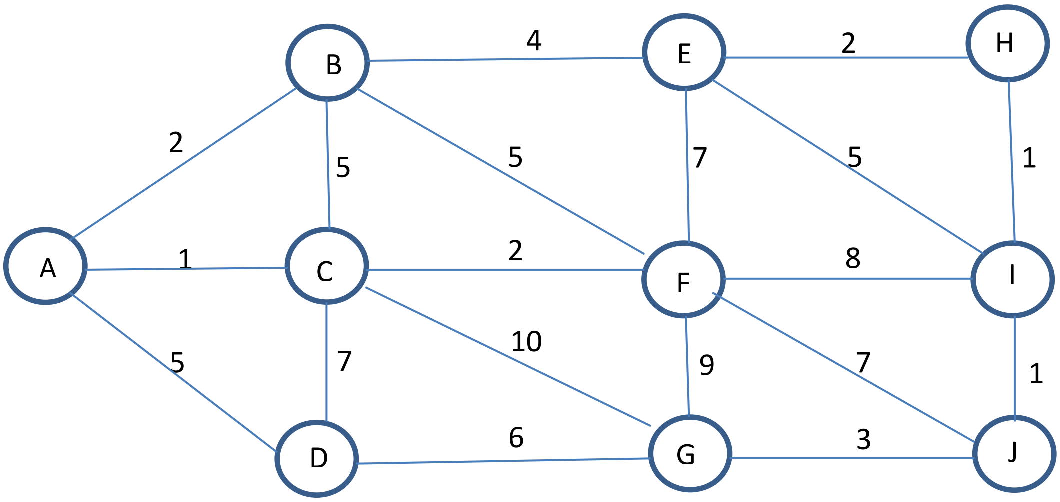
Krok 2: Algoritmus končí, pokud máme propojeny všechny vrcholy. Pokud se to nepodaří napoprvé, projíždím hrany opět od nejkratší.

Krok 3: Určíme délku minimální kostry (minimální délka kabelu je 4+3+5+10 = 22 metrů).

10.2 Nejkratší cesta v grafu (Dijkstrův algoritmus)

Používá se k nalezení nejkratší vzdálenosti z počátečního místa do nějakého finálního místa. Pro výpočet se používá Dijkstrův algoritmus.

Příklad 1: Pomocí Dijkstrova algoritmu nalezněte nejkratší cestu z místa A do místa J.



Řešení:

Krok 1: Vytvoříme si tabulku, kde do řádku dáme všechny vrcholy A, B, C,..., J. Nad nimi vytvoříme řádek pro vzdálenosti. Nad počáteční vrchol A do tabulky dáme vzdálenost 0 (vzdálenost z A do A je nulová).

vzdálenost	0									
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**

Určíme vzdálenost z počátečního vrcholu A do všech uzlů B, C, D, kde se dá z vrcholu A dostat a zapíšeme do tabulky. Vybereme vrchol C s nejmenší vzdáleností a přejdeme v tabulce do tohoto bodu, nad něj napíšeme vzdálenost 1.

vzdálenost	0		1							
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)									
	C(1)									
	D(5)									

Krok 2: Určíme vzdálenost z vrcholu C do všech uzlů B, D, F, G kde se dá z vrcholu C dostat a zapíšeme do tabulky. Bod A už neuvažujeme, protože jsme v něm už byli. Pozor! Ke vzdálenosti z C do všech uzlů musíme již připočítat vzdálenost 1, kterou jsme ujeli předtím z A do C. Vybereme **z celé dosavadní tabulky** vrchol s nejmenší vzdáleností - B(2) a přejdeme v tabulce do tohoto bodu B, nad něj napíšeme vzdálenost 2.

vzdálenost	0	2	1							
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)		B(6)							
	C(1)		D(8)							
	D(5)		F(3)							
			G(11)							

Krok 3: Určíme vzdálenost z vrcholu B do všech uzlů E, F, kde se dá z vrcholu B dostat, a zapíšeme do tabulky. Body A a C neuvažujeme, protože jsme v nich už byli. Pozor! Ke vzdálenosti z B do všech uzlů musíme už připočítat vzdálenost 2, kterou jsme ujeli předtím z A do B. Vybereme **z celé dosavadní tabulky** vrchol s nejmenší vzdáleností - F(3) a přejdeme v tabulce do tohoto bodu, nad něj napíšeme vzdálenost 3.

vzdálenost	0	2	1			3				
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)	E(6)	B(6)							
	C(1)	F(7)	D(8)							
	D(5)		F(3)							
			G(11)							

Krok 4: Postupujeme stejným způsobem dále krok po kroku.

vzdálenost	0	2	1	5		3				
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)	E(6)	B(6)			E(10)				
	C(1)	F(7)	D(8)			I(11)				
	D(5)		F(3)			J(10)				
			G(11)			G(12)				

V tabulce výše už jsme sice došli do konečného vrcholu J, ale ve zbytku tabulky jsou ještě kratší možnosti trasy – pokračujeme ještě dále, abychom zjistili, zda není možnost se dostat do J kratší cestou.

vzdálenost	0	2	1	5	6	3				
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)	E(6)	B(6)	G(11)		E(10)				
	C(1)	F(7)	D(8)			I(11)				
	D(5)		F(3)			J(10)				
			G(11)			G(12)				

vzdálenost	0	2	1	5	6	3		8		
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)	E(6)	B(6)	G(11)	H(8)	E(10)				
	C(1)	F(7)	D(8)		I(11)	I(11)				
	D(5)		F(3)			J(10)				
			G(11)			G(12)				

vzdálenost	0	2	1	5	6	3		8	9	
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)	E(6)	B(6)	G(11)	H(8)	E(10)		I(9)		
	C(1)	F(7)	D(8)		I(11)	I(11)				
	D(5)		F(3)			J(10)				
			G(11)			G(12)				

vzdálenost	0	2	1	5	6	3		8	9	10
vrchol	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
	B(2)	E(6)	B(6)	G(11)	H(8)	E(10)		I(9)	J(10)	
	C(1)	F(7)	D(8)		I(11)	I(11)				
	D(5)		F(3)			J(10)				
			G(11)			G(12)				

Krok 5: Nyní vyšla nejkratší vzdálenost právě do konečného vrcholu J, ukončíme algoritmus a do tabulky napíšeme nad J vzdálenost 10. Tato hodnota určuje výslednou nejkratší vzdálenost z A do J.

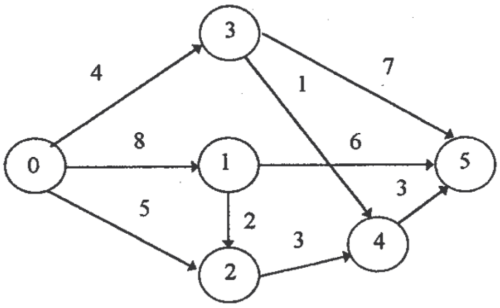
Krok 6: Sestavíme nejkratší cestu. Postupujeme zpět směrem od konečného města J k počátečnímu městu A a sledujeme, z kterého města jsme se do následujícího dostali (existují 2 alternativní řešení). Nakonec otočíme pořadí, abychom jeli z A do J.

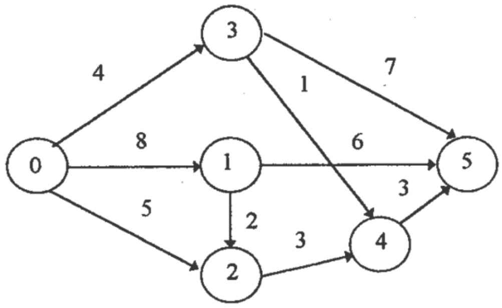
Možnost 1: J-I-H-E-B-A = 10 km **(otočíme do výsledku na A-B-E-H-I-J = 10 km)**
Možnost 2: J-F-C-A = 10 km **(otočíme do výsledku na A-C-F-J = 10 km)**

10.3 Maximální tok v síti (Ford-Fulkersonův algoritmus)

Používá se k nalezení maximálního toku, který můžeme pustit do dané sítě (např. potrubí, síť silnic). Pro výpočet se používá Ford-Fulkersonův algoritmus.

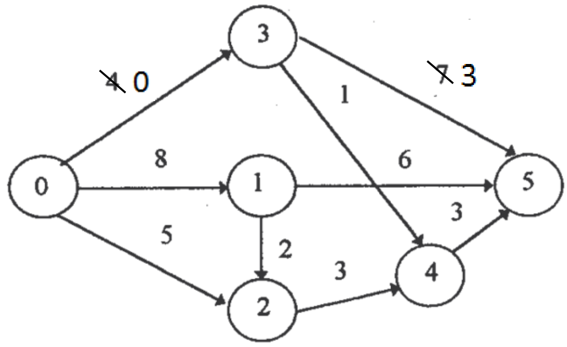
Příklad 1: Pomocí Ford-Fulkersonova algoritmu nalezněte maximální tok v síti





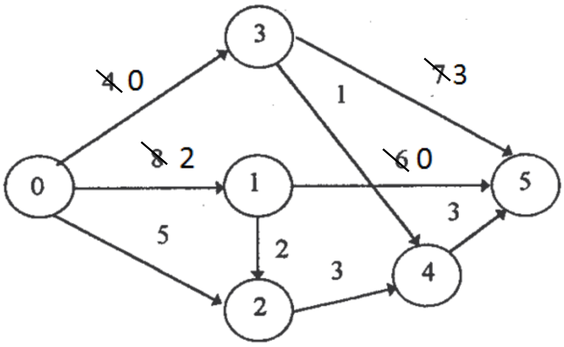
Řešení:

Krok 1: Postupujeme například vždy shora od výše položených cest. Vybereme si nějakou nenasycenou cestu (tedy cestu, která má kapacity všech hran větší než 0) např. 0,3,5. V této cestě najdeme „nejužší“ místo, v našem případě 4. Tato hodnota je rovna maximálnímu toku, který můžeme pustit po této cestě. Na uvedené cestě musíme ještě snížit kapacitu všech hran o tuto hodnotu 4.



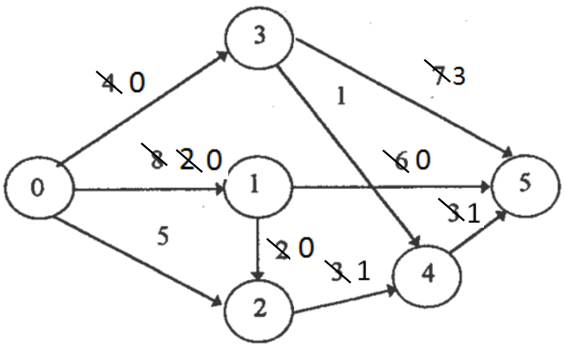
Hodnota toku prošlého po cestě 0,3,5 je T1 = 4, celková hodnota toku v celé síti je T = T1 = 4.

Krok 2: Vybereme si shora další nenasycenou cestu 0,1,5. V této cestě opět najdeme „nejužší“ místo, v našem případě 6. Na uvedené cestě musíme snížit kapacitu všech hran o hodnotu 6.



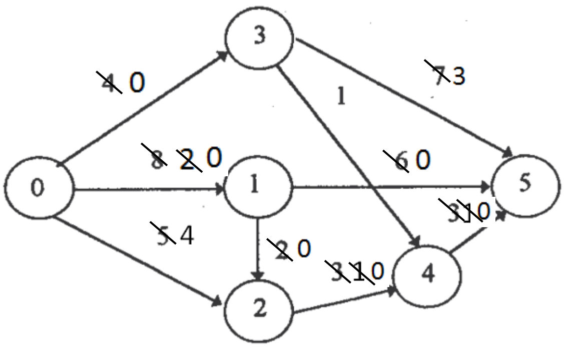
Hodnota toku prošlého po cestě 0,1,5 je T2 = 6, celková hodnota toku v celé síti je T = T1 + T2 = 4 + 6 = 10.

Krok 3: Vybereme shora další nenasycenou cestu 0,1,2,4,5. V této cestě opět najdeme „nejužší" místo, v našem případě 2. Na uvedené cestě musíme ještě snížit kapacitu všech hran o hodnotu 2.



Hodnota toku prošlého po cestě 0,1,2,4,5 je T3 = 2, celková hodnota toku v celé síti je T = T1 + T2 + T3 = 4 + 6 + 2 = 12.

Krok 4: Zbývá poslední nenasycená cesta 0,2,4,5. V této cestě opět najdeme „nejužší" místo, v našem případě 1. Na uvedené cestě snížíme kapacitu všech hran o hodnotu 1. Nyní už v dané síti nenajdeme žádnou nenasycenou cestu a postup je u konce.



Hodnota toku prošlého po cestě 0,2,4,5 je T4 = 1.

Celková hodnota toku v celé síti je T = T1 + T2 + T3 + T4 = 4 + 6 + 2 + 1 = 13.

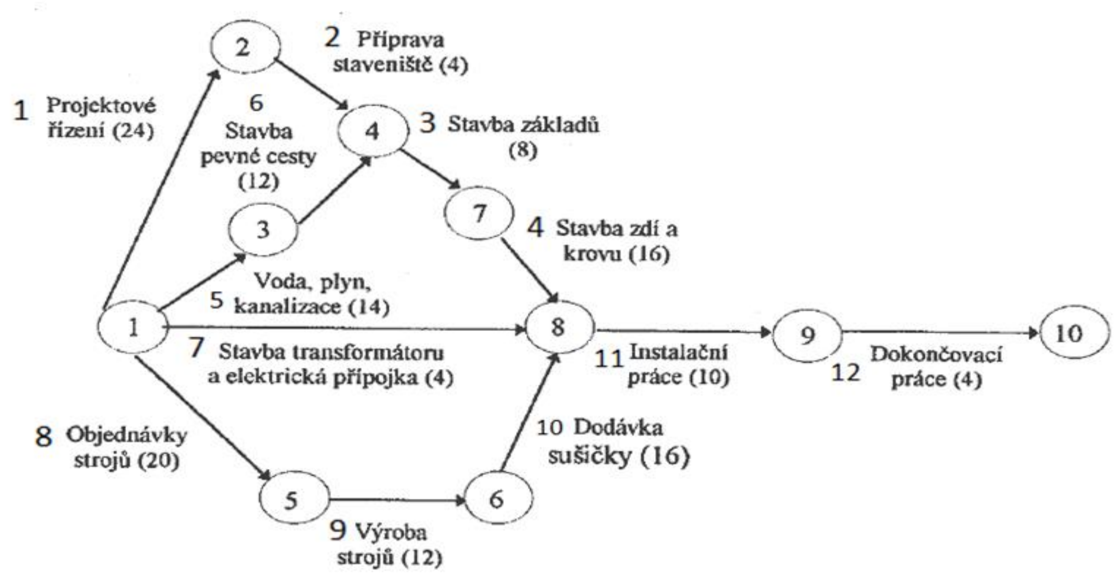
11. Projektové řízení – metoda CPM

Postup při sestavování sítě:

1) **Nejdříve vyhledáme tzv. počáteční činnosti – činnosti, kterým žádná činnost nepředchází. V našem případě činnosti 1,5,7,8. Tyto činnosti budou vycházet z počátečního uzlu sítě.**
2) **Následně vyhledáme činnosti, po kterých již žádná činnost nenásleduje. V našem případě činnost 12. Tato činnost bude končit v koncovém uzlu sítě.**
3) **Postupujeme podle tabulky činností a na počáteční činnosti postupně napojujeme další činnosti.**
4) **Očíslujeme uzly sítě tak, aby pro každou hranu sítě (i,j) platil vztah i < j. Tedy číslo uzlu na začátku činnosti musí být menší než číslo uzlu na konci činnosti:**

Příklad 1: Máme tabulku činností pro stavbu sušičky, úkolem je sestavit síťový graf

Číslo práce	Dílčí práce	Následuje práce	Doba trvání v týdnech
1	Projektové řízení výstavby haly	2	24
2	Příprava staveniště	3	4
3	Stavba základů budovy	4	8
4	Výstavba obvodových zdí, příček, krovů	11	16
5	Přípojka vody, plynu, kanalizace	6	4
6	Vybudování pevné cesty	3	12
7	Stavba transformátoru a el. přípojky	11	4
8	Objednávka strojů	9	20
9	Výstavba strojového zařízení	10	12
10	Dodávka sušičky	11	16
11	Vnitřní instalace	12	10
12	Kolaudace a dokončovací práce	-	4

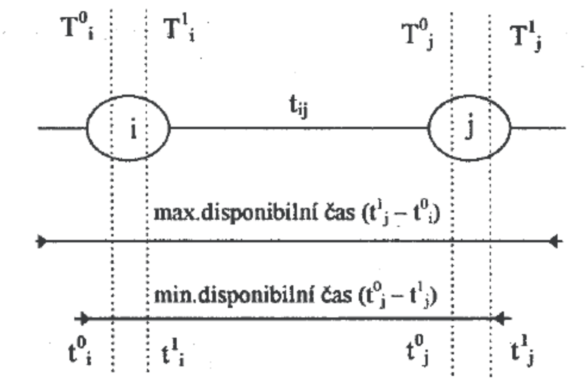


Metoda CPM

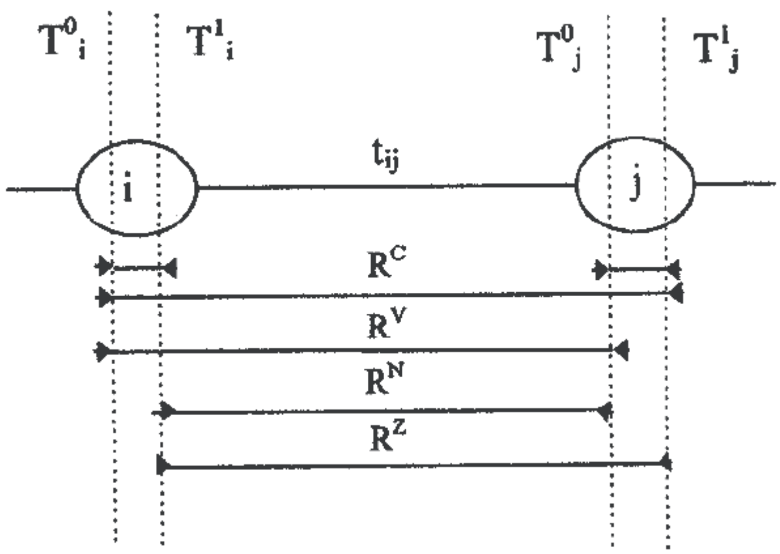
Metoda CPM (Critical Path Method) pracuje s deterministickým (nenáhodným) modelem. Základním úkolem je najít v daném modelu tzv. kritickou cestu – tedy cestu, která má nulové časové rezervy a na kterou se v projektu čeká.

Základní charakteristiky síťového diagramu jsou:

t_{ij} - doba trvání činnosti (i, j)
t^0_i - termín nejdříve možného zahájení činnosti (i, j)
t^0_j - termín nejdříve možného ukončení činnosti (i, j)
t^1_i - termín nejpozději možného zahájení činnosti (i, j)
t^1_j - termín nejpozději přípustného ukončení činnosti (i, j)
T^0_i- termín nejdříve možného výskytu počátečního uzlu činnosti (i, j)
T^0_j- termín nejdříve možného výskytu koncového uzlu činnosti (i, j)
T^1_i- termín nejpozději přípustného výskytu počátečního uzlu činnosti (i, j)
T^1_j- termín nejpozději přípustného výskytu koncového uzlu činnosti (i, j)



Přehled časových rezerv



Celková časová rezerva $R^c_{ij} = T^1_j - T^0_i - t_{ij}$

Volná časová rezerva $R^V_{ij} = T^0_j - T^0_i - t_{ij}$

Nezávislá časová rezerva $R^n_{ij} = T^0_j - T^1_i - t_{ij}$

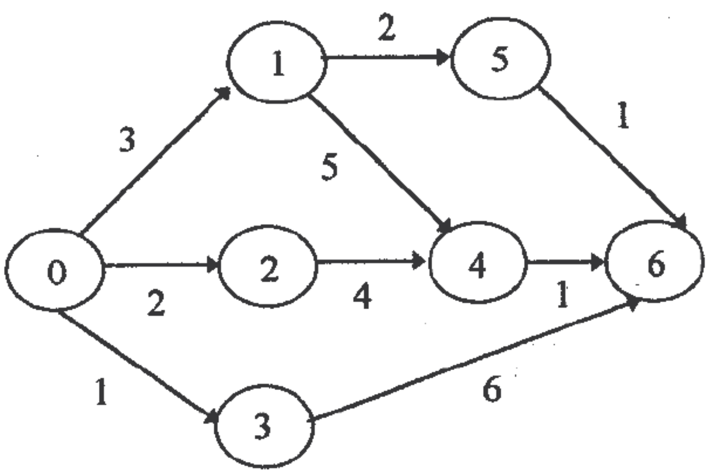
Zvláštní časová rezerva $R^z_{ij} = T^1_j - T^1_i - t_{ij}$

Interferenční (kritická rezerva uzlu) $R_i = T^1_i - T^0_i$

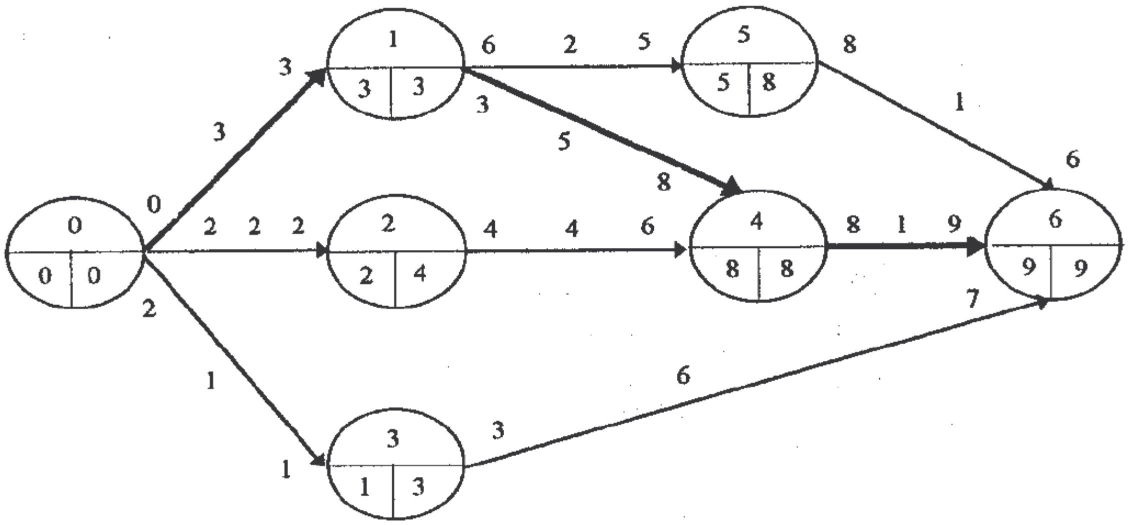
Příklad 2: Máme tabulku činností, sestavte síťový graf, najděte kritickou cestu a vypočítejte rezervy pro všechny činnosti.

Číslo činnosti	Navazuje na činnost (Předcházející činnost)	Doba trvání v týdnech
0 1	-	3
0 2	-	2
0 3	-	1
1 5	0 1	2
1 4	0 1	5
2 4	0 2	4
4 6	1 4 2 4	1
5 6	1 5	1
3 6	0 3	6

Krok 1: Z tabulky činností vytvoříme síťový graf



Krok 2: Rozdělíme uzly na 3 části, viz obrázek a do prvního uzlu dáme vždy 0 nalevo.



Krok 3: Přičítáme k levé části uzlu střed šipky (délku trvání činnosti) a výsledek píšeme na konec šipky. Pokud vstupuje do uzlu více šipek, píšeme do levé části uzlu vždy NEJVĚTŠÍ číslo z konců šipek.

Krok 4: Číslo nalevo v posledním uzlu je **délka trvání projektu**, tedy jeden z výsledků úlohy. Toto číslo přepíšu do pravé části uzlu a postupuji pozpátku – viz dále.

Krok 5: Odečítáme od pravé části uzlu střed šipky (délku trvání činnosti) a výsledek píšeme na začátek šipky. Pokud vystupuje z uzlu více šipek, píšeme do pravé části uzlu vždy NEJMENŠÍ číslo ze začátků šipek.

Krok 6: Pokud jsme postupovali správně, musí nám vyjít v pravé části prvního uzlu opět 0.

Krok 7: **Kritická cesta** vede přes činnosti a uzly, kde se konec předchozí šipky rovná začátku následující šipky (tedy přes uzly 0,1,4,6). Všechny časové rezervy viz dále jsou tedy v kritické cestě nulové

Krok 8: Pokud by byla v grafu tzv. fiktivní činnost (čas trvání fiktivní činnosti je 0), kritická cesta by se nezměnila.

Krok 9: Dopočítáme rezervy pro jednotlivé činnosti (všimněme si, že rezervy na kritické cestě jsou všechny nulové ☺)

		Rezerva celková	**Rezerva volná**	**Rezerva nezávislá**	**Rezerva zvláštní**
Činnost	**Doba trvání činnosti**	Rc	Rv	Rn	Rz
0,1	3	0	0	0	0
0,2	2	2	0	0	2
0,3	1	2	0	0	2
1,4	5	0	0	0	0
1,5	2	3	0	0	3
2,4	4	2	2	0	0
3,6	6	2	2	0	0
4,6	1	0	0	0	0
5,6	1	3	3	0	0

Domácí úkol☺

Projekt je dán výčtem činností:

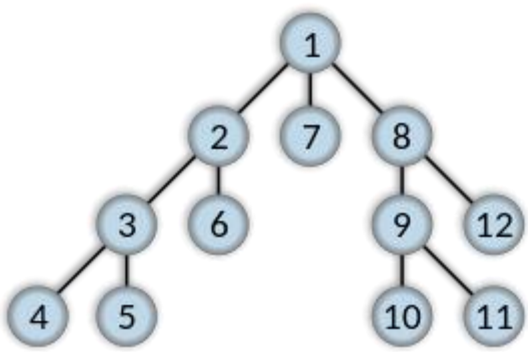
Činnost	Předcházející činnost	Délka trvání
1 2	-	2
1 3	-	4
2 4	1 2	5
2 3	1 2	7
3 4	1 3 2 3	6
4 5	2 4 3 4	1

a) Nakreslete síťový graf
b) Určete kritickou dobu projektu (délku projektu)
c) Najděte a v grafu označte kritickou cestu
d) Jak se změní kritická cesta doplněním fiktivní hrany 3 5?

Prohledávání do hloubky a do šířky

Prohledávání do hloubky

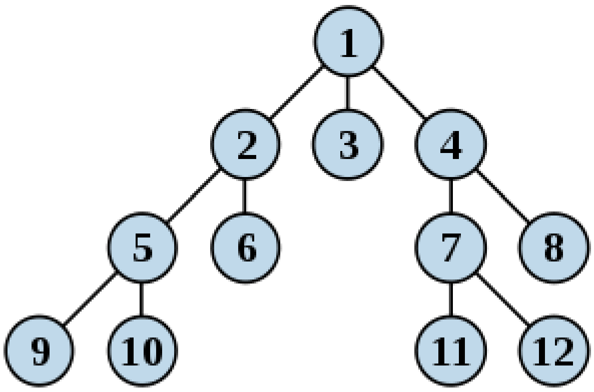
Od prvního vrcholu postupujeme vždy k dalšímu vrcholu podle jeho čísla co nejvíce „do hloubky“, pokud to jde. Pokud narazím na vrchol, z nějž už nelze dále pokračovat, vracím se zpět tzv. backtrackingem – viz doučování.



Pořadí v jakém je přistupováno k vrcholům při prohledávání do hloubky

Prohledávání do šířky

Algoritmus nejprve projde všechny sousedy startovního vrcholu, poté sousedy sousedů atd až projde celou množinu vrcholů – viz doučování.



Pořadí v jakém je přistupováno k vrcholům při prohledávání do šířky

Příklad: Je dána matice sousednosti neorientovaného neohodnoceného grafu (hrany jsou v matici označeny jedničkami)

Rozhodněte, zda je graf souvislý, a pokud není, určete, které uzly tvoří jeho jednotlivé komponenty souvislosti (souvislé části)

a) prohledáváním grafu do hloubky
b) prohledáváním grafu do šířky

V obou případech začněte prohledávání nové komponenty vždy v uzlu, který má ze všech neprojitých uzlů nejnižší pořadové číslo (prohledávání začněte tedy v uzlu v1). V případě více možností v průběhu prohledávání navštivte vždy nejdříve uzel s nižším pořadovým číslem. U obou úkolů uveďte uzly jednotlivých komponent v pořadí, v jakém byly projity při prohledávání a znázorněte graficky.

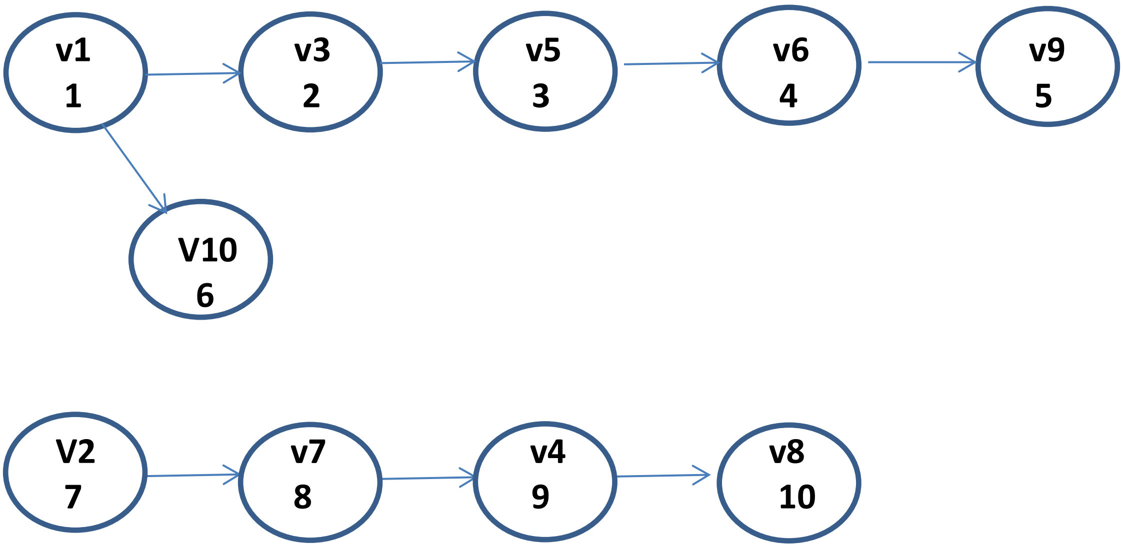
	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
V1	-	-	1	-	-	-	-	-	1	1
V2	-	-	-	-	-	-	1	1	-	-
V3	1	-	-	-	1	1	-	-	-	-
V4	-	-	-	-	-	-	1	1	-	-
V5	-	-	1	-	-	1	-	-	1	-
V6	-	-	1	-	1	-	-	-	1	-
V7	-	1	-	1	-	-	-	-	-	-
V8	-	1	-	1	-	-	-	-	-	-
V9	1	-	-	-	1	1	-	-	-	-
V10	1	-	-	-	-	-	-	-	-	-

Řešení:

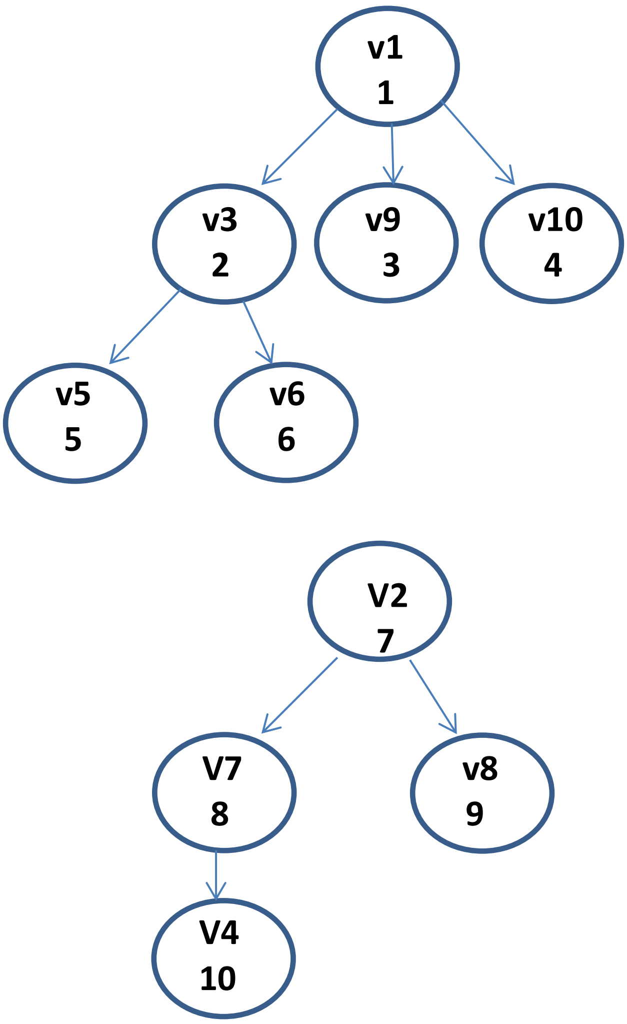
Souvislý graf je takový graf, v němž platí, že pro každé dva vrcholy x, y existuje sled z x do y

Graf není souvislý, nelze se dostat z libovolného vrcholu do libovolného vrcholu (např. z v1 do v2)

a) **prohledáváním grafu do hloubky**



b) **prohledáváním grafu do šířky**



Ukázkové testové příklady k procvičení

Příklad 1: Pro ceremoniální účely bylo třeba vyrobit větší množství šperků. Vzhledem

k nedostatku času nebylo možné zadat výrobu jednomu klenotníkovi, a proto se zakázka rozdělila mezi 5 klenotníků po jednotlivých produktech. V tabulce je odhad ceny v tis. Kč za dokončení každé části zakázky jednotlivými klenotníky.

	Korunka	Náhrdelník	Náušnice	Prsteny	Náramky
Klenotník 1	42	35	19	14	20
Klenotník 2	35	33	16	14	21
Klenotník 3	45	33	15	12	20
Klenotník 4	38	32	19	16	21
Klenotník 5	29	39	17	19	23

Určete jak zakázku rozdělit, aby byly klenoty vyrobeny s minimálními náklady.

Příklad 2: Podnik rozváží chlévskou mrvu z objektů živočišné výroby na hony. Vzdálenosti a

kapacity jsou uvedeny v tabulce. Úkolem je navrhnout plán přepravy, při kterém vozidla ujedou minimální počet tunokilometrů.

kapacity	požadavky	matice vzdáleností		
225	400	2,0	1,4	1,0
175	125	1,9	1,2	1,2
700	450	1,0	1,7	2,0

a) Proveďte vyvážení úlohy
b) Najděte výchozí řešení – napište, kterou metodu jste provedli
c) Najděte optimální řešení
d) Vypočítejte hodnotu účelové funkce pro optimální řešení
e) Určete perspektivitu a propustnost alespoň dvou neobsazených a dvou obsazených spojů

Příklad 3: Je dáno výchozí a optimální řešení modelu lineárního programování

		53	5	25	0	0	0	-100		
C_B	B	x1	x2	x3	x4	x5	x6	x7	b	Ω
0	x_4	1	1	1	1	0	0	0	450	450
0	x_5	1	0	0	0	1	0	0	100	100
-100	x_7	6	1	3	0	0	-1	1	1300	650/3
zj-cj		-653	-105	-325	0	0	100	0	-130000	
0	x_6	0	2	0	3	3	1	-1	350	
53	x_1	1	0	0	0	1	0	0	100	
25	x_3	0	1	1	1	-1	0	0	350	
zj-cj		0	20	0	25	28	0	100	14050	

a) Které jsou strukturní (rozhodovací), doplňkové a pomocné (umělé) proměnné
b) Které jsou základní a nezákladní proměnné
c) Vypsat matici B^{-1} a B

d) Z výsledné tabulky sestavit vektor bazického (základního) a obecného řešení
e) V jakém rozsahu lze zařadit x_2 aniž by došlo ke změně báze
f) V jakém rozsahu lze měnit c_2 aniž by došlo ke změně báze
g) Jak se změní optimální řešení při zařazení 10 jednotek x_2
h) Udělat první krok simplexového algoritmu a určit, jaká bude změna účelové funkce v kroku bezprostředně následujícím po výchozím řešení

Příklad 4: V následující tabulce jsou uvedeny činnosti projektu a doby jejich trvání.

a) nakreslete síťový graf a správně očíslujte uzly
b) zjistěte minimální dobu do ukončení celého projektu
c) určete kritickou cestu
d) určete všechny typy rezerv všech činností

Činnost	Následující činnosti	Doba trvání
A	C	7
B		1
C	B, D	7
D	E	1
E		7

Příklad 5: Nalezněte graficky maximum funkce z = 3.x1 + 4.x2 za podmínek:

$-2x1 + x2 \leq 2$

$x1 + 5x2 \leq 35$

$3x1 - 4x2 \leq 12$

$3x1 + 4x2 \geq 24$

$x1 \geq 0,\ x2 \geq 0$

Odhadněte z grafu optimální hodnoty všech proměnných a účelové funkce.