

Naloga 5.2

5.2.1

Ena možna rešitev je:

Miha:Petra
Lojze:Ana
Tomaž:Melita
Janez:brez
Jerneja:brez

Druga:

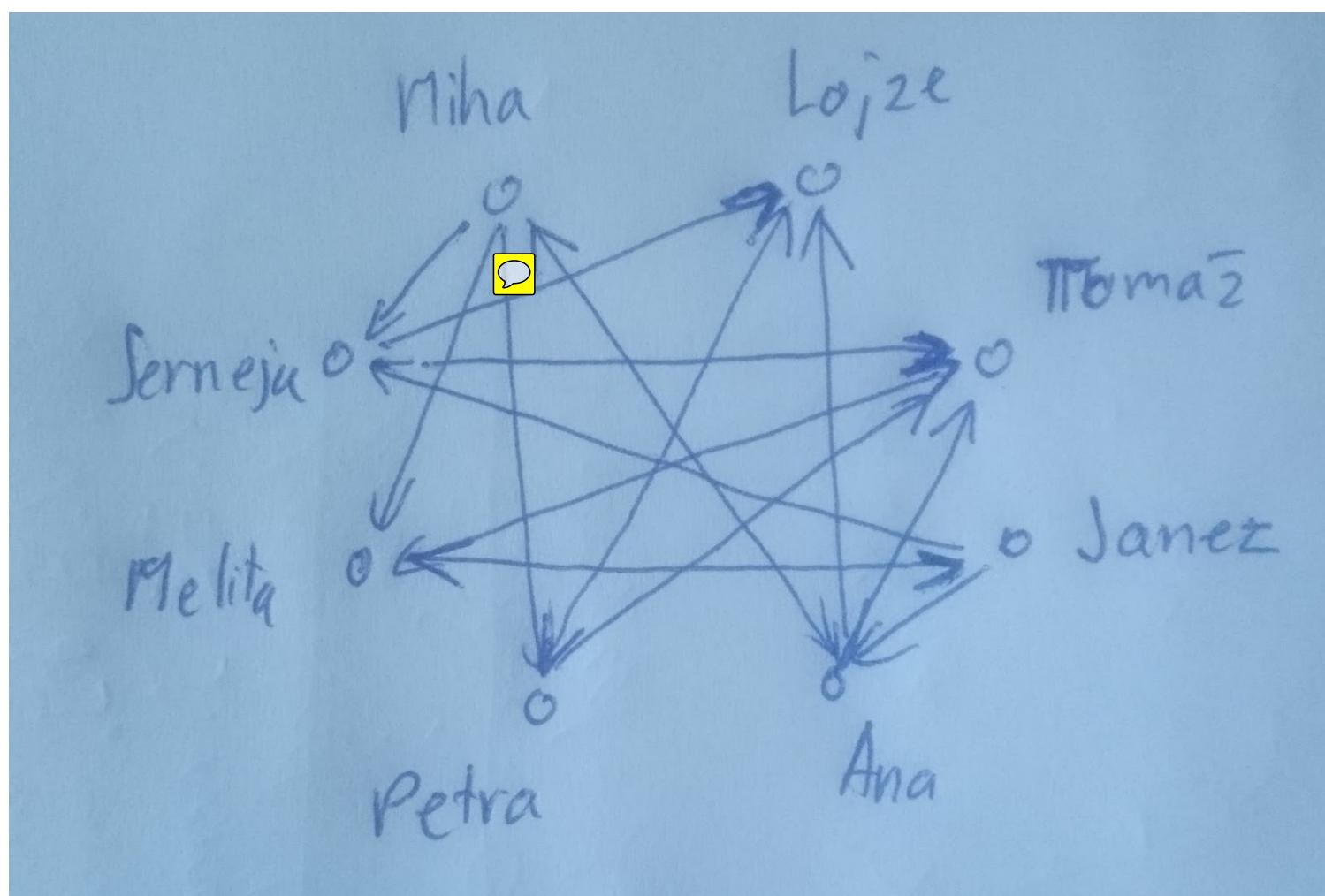
Miha:Ana
Lojze:Petra
Tomaž:brez
Janez: Melita
Jerneja:brez

Sestaviti se da še dve tako da Tomaža in Janeza zamenjamo pri prvi in drugi rešitvi.

Vseh se ne da poparčkati.

5.2.2

Enosmerna povezava pomeni zanimanje ene osebe za drugo ne pa tudi obratno. Dvosmerna pomeni obojestransko zanimanje. Glede na definicijo trdnosti so smiselne le dvosmerne povezave.



5.2.3

```
sort(i=>G.V[i],G.V[i].E.size(),ASC)
while countremoved!=G.V.size():
    for V in G.V:
        for i in range(V.E.size()):
            if !V.E[i].removed:
                V.E[i].removed=1
                countremoved+=1
                out.add(V.name+" : "+V.E[i].name)
                break
            V.removed=1
            countremoved+=1|
```

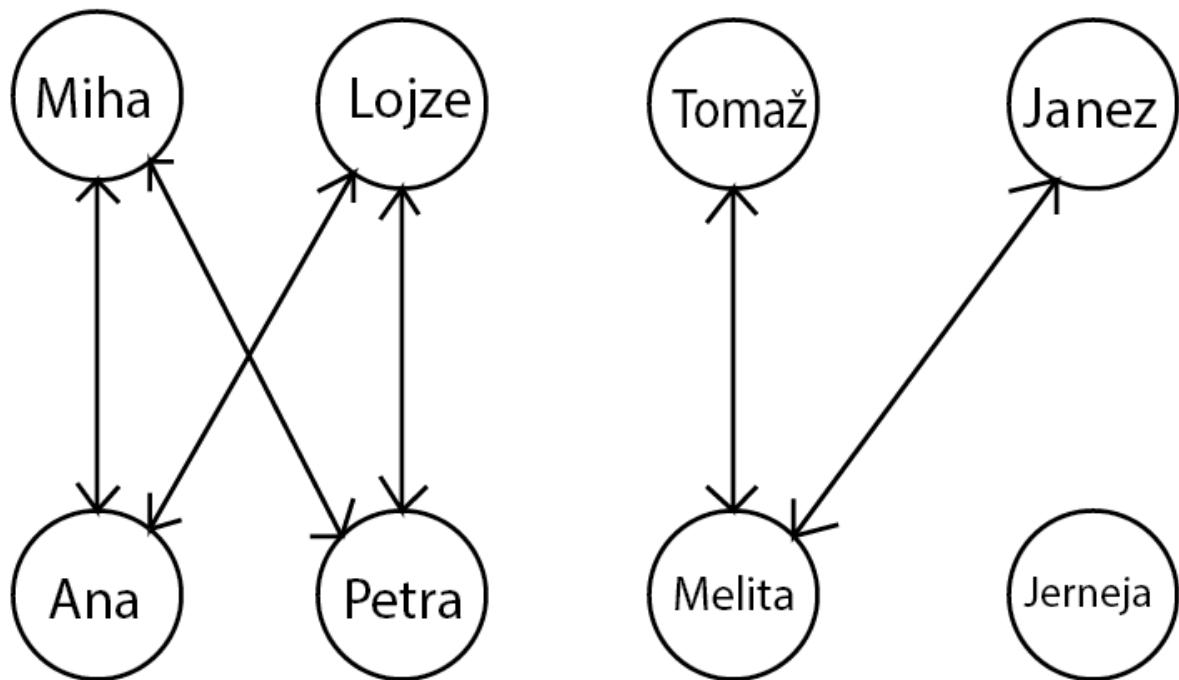
Graf vsebuje samo dvosmerne povezave. Enosmerne niso smiselne zato jih ne dodamo. Graf najprej sortiramo tako da so vozl. z najmanj dvosmernimi povezavami na začetku. S tem zagotovimo da bolj ozko usmerjenim ne odvzamemo možnosti (tako dobimo več parov).



5.2.1

Miha : Ana
Lojze : Petra
Tomaž : Melita
Zakon Janeza in Jerneje pa ni trden.

5.2.2



Vozlišča so osebe, povezave pa predstavljajo naklonjenost partnerju. V graf dodamo le povezave, ki potekajo v obe smeri. Na primer od Tomaža do Melite in obratno. Tako bomo iz grafa razbrali vse možnosti za trdne zakonske zveze.

5.2.1 Za naslednje osebe imamo podane možne partnerje:

- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

Miha Ana,

Lojze Petra,

Tomaž Melita,

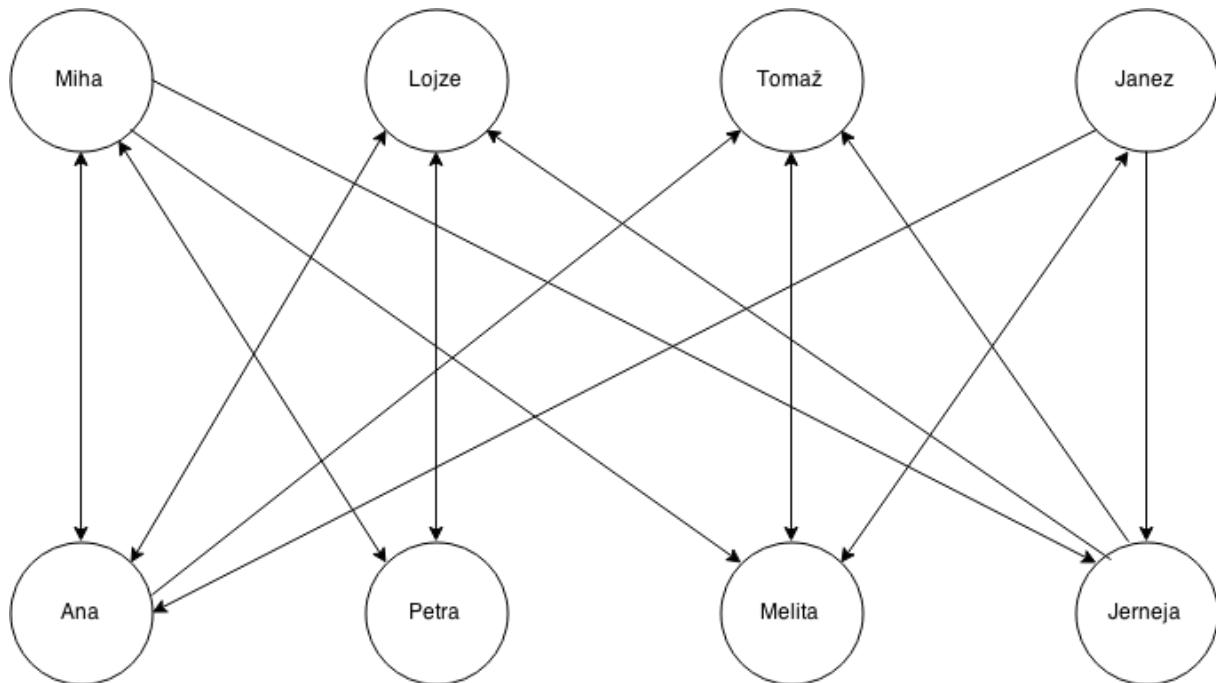
Janez Jerneja 

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

Vozlišča so osebe, povezave pa predstavljajo možne partnerje neke osebe.

Glede na definicijo trdnosti so smiselne samo povezave od neke osebe do oseb, katere so možni pari (se pravi, povezav, ki ne predstavljajo možnih partnerjev ne rišemo).

Graf:



5.2.3 Opišite algoritmom, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru?
(Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

Algoritom začne v naključnem vozlišču ter preišče možne partnerje te osebe, dokler ne najde osebe, katera ima prav tako prvo osebo za partnerja. Nato ti dve osebi združi v par ter ju odstrani iz grafa. Če ne najde nobene take osebe, samo nadaljuje z iskanjem pri naslednji osebi.

To ponavlja dokler ne odstrani vseh vozlišč, tako, da v grafu ni več dvosmernih povezav.

Nato pregleda vse ostale osebe. Pri prvi osebi vzame prvo oseba, katera je možen par, ter ju zdruči v par in odstrani. To ponavlja dokler v grafu ni več vozlišč.



Naloga 5.2

Problem trdne zakonske zveze. V matematiki in računalništvu je znan problem trdne zakonske zveze (angl. *stable marriage*). Pri tem problemu imata vsak fant in vsako dekle preferenco, s kom bi se rad oziroma rada poročila ter želimo poroke določiti tako, da so trdne. Poroka med fantom F in dekletom D je trdna, če **hkrati ne velja**:

- da obstaja dekle D', ki bi jo fant F imel raje kot dekle D; in
- da ne obstaja fant F', ki bi ga imelo dekle D raje kot fanta F.

Z drugimi besedami, želimo dobiti takšne pare, da ne obstajata nobeno dekle in fant, za katera bi veljalo, da bi se imela vzajemno raje, kot pa imata rada partnerja v trenutni poroki.

Za obvezni del naše naloge bomo problem nekoliko **poenostavili**. Tako bo preferenca zgolj binarno število 1 - se želi poročiti z osebo, ali 0 - se ne želi poročiti z osebo. Zakon velja za trdnega, če se tako F želi poročiti z D kot tudi obratno. Naloga je, da poiščete čim več trdnih zakonov.

Vprašanja:

5.2.1 Za naslednje osebe imamo podane možne partnerje:

- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

5.2.3 Opišite algoritem, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru? (Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

5.2.4 (Dodatna) Končno rešujemo še splošni problem trdne zakonske zveze. V tem primeru preference predstavljajo rangiranje partnerjev glede na željo po poroki posamezne osebe. Predlagajte rešitev in utemeljite njeno pravilnost. (Namig: V resnici pri reševanju splošnega problema ne uporabimo modeliranja z grafom.)

5.2.1 Za naslednje osebe imamo podane možne partnerje:

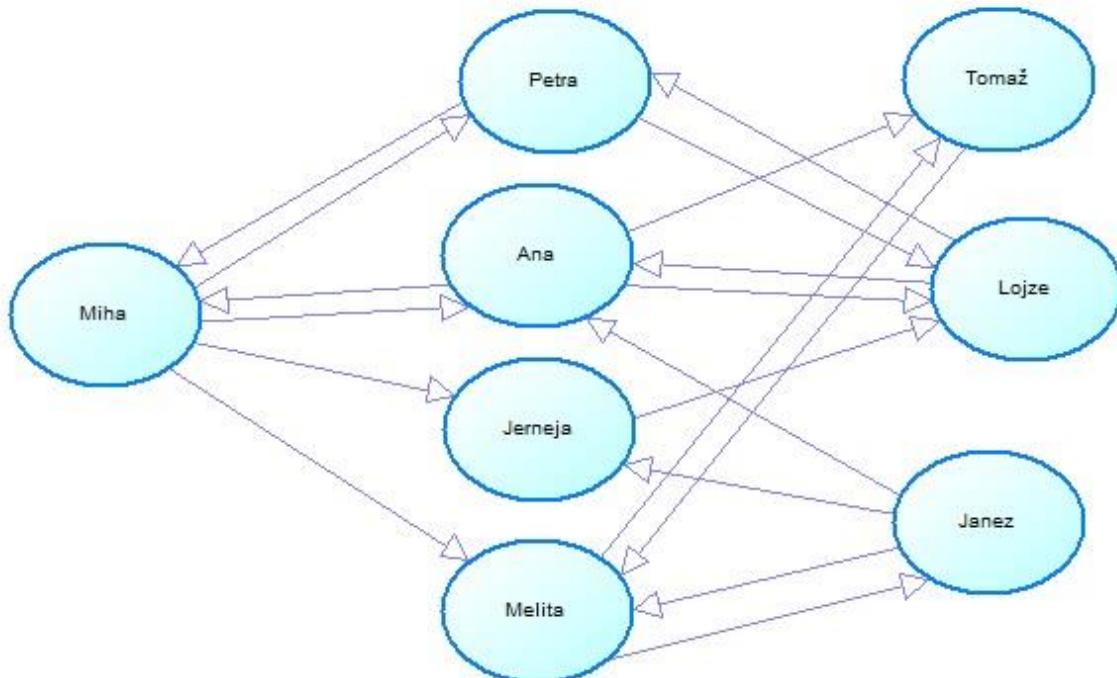
- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

1. Rešitev: Miha - Ana, Lojze - Petra, Janez - Melita
2. Rešitev: Miha - Petra, Lojze - Ana, Melita - Janez
3. Rešitev: Miha - Petra, Lojze - Ana, Janez - Melita
4. Rešitev: Miha - Ana, Lojze - Petra, Melita - Tomaž

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

Problem bi predstavil z usmerjenim grafom, kjer bi enostranska povezava predstavljal možnost zveze, obojestranska povezava pa možnost trdne zveze.



5.2.3 Opišite algoritem, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru? (Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

	Miha	Lojze	Janez	Tomaž
Ana	1	1		
Petra	1	1		
Melita			1	1
Jerneja				

V tabelo vstavimo enice, kjer so obojestranske povezave. Če eno ime vsebuje več enic, potem eno izberemo naključno. V našem primeru bi dobili pare: Miha – Ana, Lojze – Petra, Tomaž – Melita.

5.2.4 (Dodatna) Končno rešujemo še splošni problem trdne zakonske zveze. V tem primeru preference predstavljajo rangiranje partnerjev glede na željo po poroki posamezne osebe. Predlagajte rešitev in utemeljite njeno pravilnost. (Namig: V resnici pri reševanju splošnega problema ne uporabimo modeliranja z grafom.)

Želje moških:

Miha	Lojze	Janez	Tomaž
Ana	Petra	Melita	Melita
Petra	Ana	Ana	
Melita			
Jerneja			

Želje žensk:

Ana	Petra	Jerneja	Melita
Miha	Lojze	Tomaž	Tomaž
Lojze	Miha	Lojze	Janez
Tomaž			

V dveh tabelah sta po stolpcih rangirane želje oseb po poroki. V primeru zelene barve smatramo, da gre za trdno zvezo, za rumeno barvo, da gre za mogočo zvezo, rdeča pa pomeni nemogočo zvezo. V našem primeru smo dobili naslednje pare: Ana – Miha, Petra – Lojze, Melita – Tomaž. Težava je, Jerneja ostane brez partnerja, zato bi ji lahko uredili zmenek z Lojzetom ter upali, da zveza lahko postane mogoča.



Naloga 5.2

5.2.1

Vsaj ena možna rešitev je, da imamo naslednje pare:

- MIHA + ANA
- LOJZE + PETRA
- MELITA + TOMAŽ

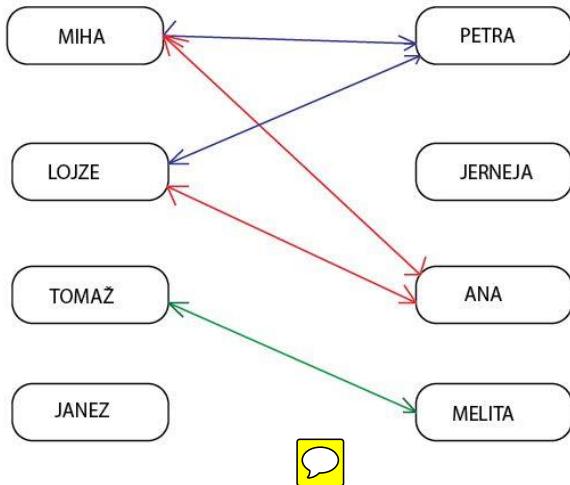
Jerneja ostane brez partnerja, saj ima Tomaž že izbranko, oziroma Jerneje sploh nima na seznamu, prav tako Lojze.

ANA	Miha	Lojze	Tomaž
PETRA	Lojze	Miha	
MELITA	Tomaž	Janez	
JERNEJA	Tomaž	Lojze	

MIHA	Ana	Petra	Melita	Jernja
LOJZE	Petra	Ana		
TOMAŽ	Melita			
JANEZ	Ana	Melita	Jerneja	

5.2.2

Na grafu sem narisal samo smiselne povezave za možnost zveze za podane osebe, povezave.



5.2.3

Algoritem bi zastavil tako, da bi v vsakem koraku našel vozlišče z najmanj povezavami in ga združil z enim izmed sosedov. Če bi v katerem od korakov, bila nova zveza močnejša od že nastale, bi slednjo izbrisal. To ponavljamo za vse osebe dokler ne bi več mogel združiti noben vozlišči.



5.2.1 Za naslednje osebe imamo podane možne partnerje:

- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

Tomaž $\leftarrow \rightarrow$ Melita

Miha $\leftarrow \rightarrow$ Ana

Lojze $\leftarrow \rightarrow$ Petra

ali pa

Janez $\leftarrow \rightarrow$ Melita

Lojze $\leftarrow \rightarrow$ Ana

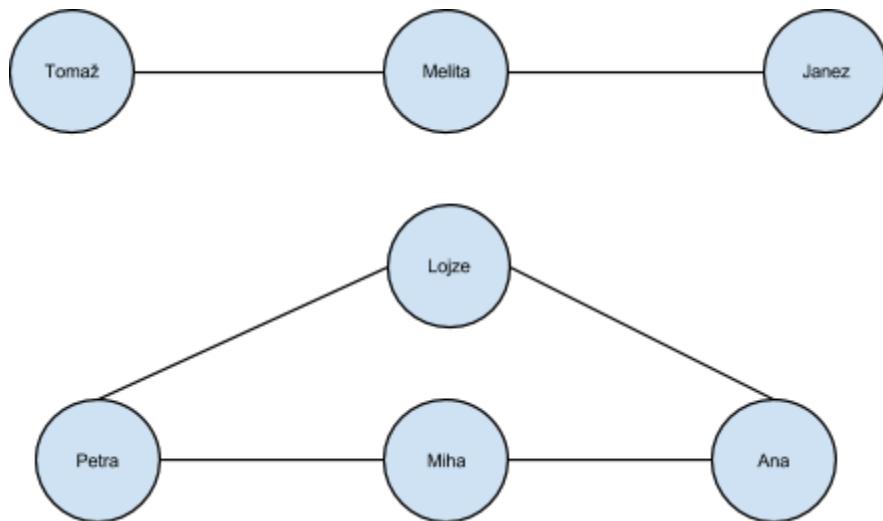
Miha $\leftarrow \rightarrow$ Petra

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

Smiselne so samo obojestranske povezave.

V spodnjem grafu so vse povezave obojestranske.

Vozlišča so osebe, povezave pa razmerja za osebe. Povezava pove kdo je komu všeč.



5.2.3 Opišite algoritmom, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru? (Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

Algoritmom deluje tako, da zgradi graf samo z obojestranskimi povezavami.

Potem je pa odvidno od tega kaj želimo, ali vse možne rešitve ali samo eno. Če želimo samo eno rešitev, algoritmom pošišče vozlišče z najnižjo stopnjo. (Stopnja vozlišča pomeni koliko povezav ima to vozlišče.)

Potem vzamemo iz grafa izbrano vozlišče in sosednje vozlišče z najnižjo stopnjo.

Če pa želimo vse možne kombinacije, potem pa uporabimo podatkovno strukturo, ki ne dovoljuje ponavljanja (v strukturo vedno shranjujemo tako, da je vedno en spol na enem mestu in drugi na drugem).

V tem primeru pa izberemo vozlišče z najvišjo stopnjo in shranimo razmerja s sosedmi v strukturo. Potem iz grafa odstranimo izbrano vozlišče in vse sosedje, ki imajo samo povezavo na to vozlišče.

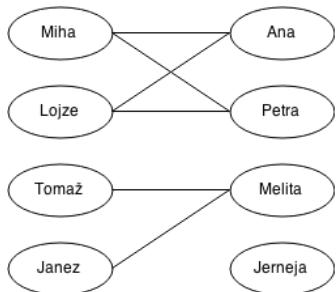
Naloga 5.2.1

Vse rešitve z 3 pari (2 osebi v vsakem primeru ostaneta brez trdega zakona):

```
{ (Miha, Ana), (Lojze, Petra), (Tomaž, Melita) }
{ (Miha, Ana), (Lojze, Petra), (Janez, Melita) }
{ (Miha, Petra), (Lojze, Ana), (Tomaž, Melita) }
{ (Miha, Petra), (Lojze, Ana), (Janez, Melita) }
```

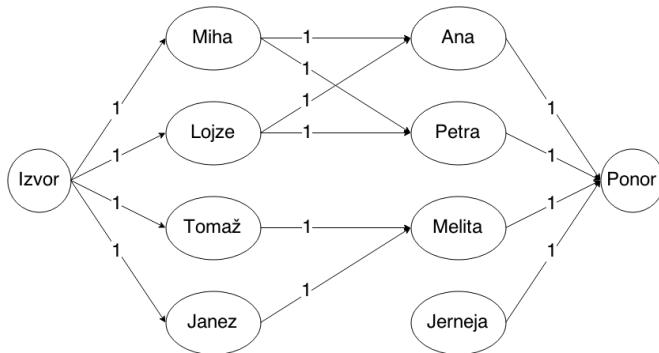
Naloga 5.2.2

Vsako osebo lahko predstavimo z vozliščem. Med vsakim parom oseb kjer bi bil zakon trden naredimo povezavo. Tak graf je dvostranski. Na levi strani so fantje, na desni pa dekleta.



Naloga 5.2.3

Tako zasnovan problem je problem ujemanja (matching). Iščemo nabor povezav brez skupnih vozlišč. Ker je graf dvodelni, lahko problem preoblikujemo v problem iskanja največjega pretoka. Vsaki povezavi nastavimo smer iz leve proti desni. Nato dodamo izvor z povezavami na vsa leva vozlišča, in ponor z povezavami iz vseh desnih. Vse povezave imajo kapaciteto 1.



Ko v takem grafu najdemo največji pretok, je vsaka povezava iz levega vozlišča na desnega ki ima pretok predstavljal veljaven par. Levo vozlišče ne bo nikoli oddalo več kot eno enoto, saj ima vsako levo vozlišče vhoda največ 1. Desno vozlišče ne bo nikoli sprejelo več kot 1 enote, saj ima izhod največ 1. To pomeni da nabor najdenih povezav ne bi imelo skupnih vozlišč.

5.2.1

Možne rešitve:

Ana – Lojze

Ana – Miha

Janez – Melita

Lojze – Petra

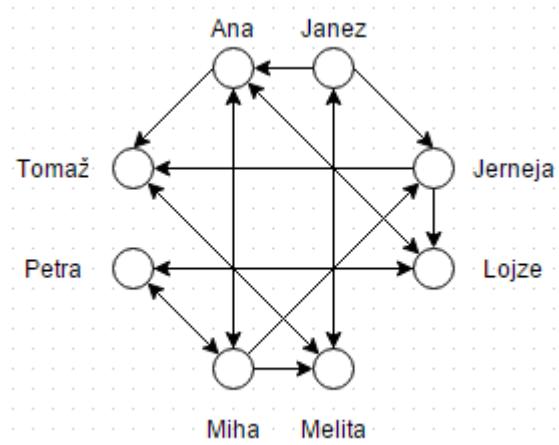
Jerneja /



Melita – Tomaž

Miha – Petra

5.2.2



5.2.1 *

Partnerji	Miha	Lojze	Tomaž	Janez
Ana	1 \ 1	1 \ 1	1 \ 0	0 \ 1
Petra	1 \ 1	1 \ 1	0 \ 0	0 \ 0
Melita	0 \ 1 ^{*opomba}	0 \ 0	1 \ 1	1 \ 1
Jerneja	0 \ 1	1 \ 0	1 \ 0	0 \ 1

Možne rešitve

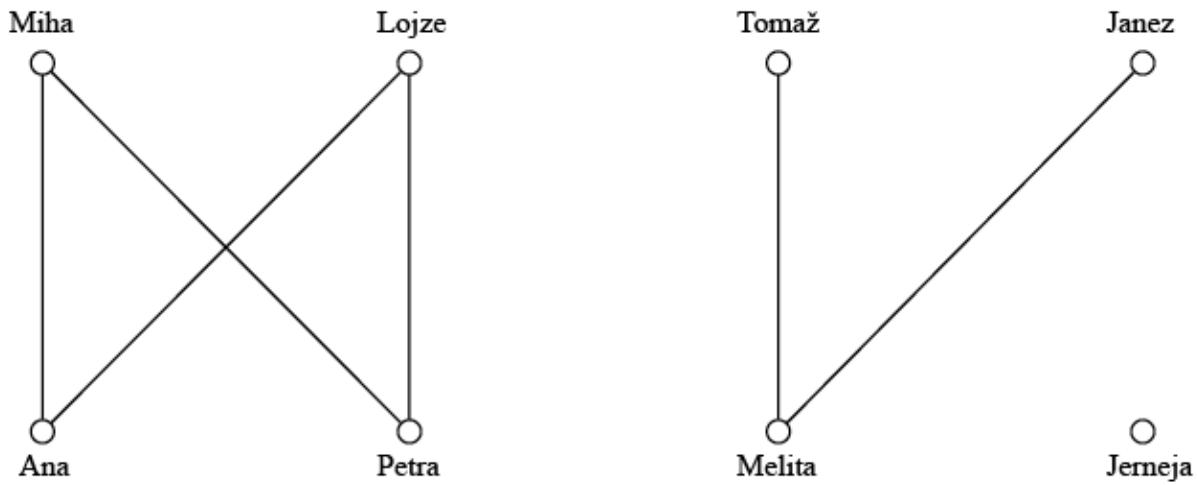
1. [[Miha, Ana], [Lojze, Petra], [Tomaž, Melita]]
2. [[Miha, Ana], [Lojze, Petra], [Janez, Melita]]
3. [[Miha, Petra], [Lojze, Ana], [Tomaž, Melita]]
4. [[Miha, Petra], [Lojze, Ana], [Janez, Melita]]

5.2.2

Problem bi z grafom predstavil tako, da bi mu dal za začetek **neusmerjene** (*Slika 1*) (ker mora za trdno zakonsko zvezo veljati, da se imata osebi medsebojno radi) povezave. Za vozlišča bi izbral imena oseb, ki jih obravnavamo. Graf se lahko poenostavi, ker imamo tabelo razdeljeno na stolpce (moški) in vrstice (ženske). Stolpci bodo v zgornjem delu grafa, vrstice pa v spodnjem delu grafa. Za dodatno poenostavitev grafa damo povezave od nekega vozlišča »stolpec« do drugega vozlišča »vrstica« le, če ima to drugo vozlišče »vrstica« tudi povezavo v prvo vozlišče »stolpec«. Torej so smiselne le povezave, kjer se imata ti dve osebi medsebojno radi. Jerneja ima rada Lojzeta in Tomaža, vendar tu ne velja obratno, zato to vozlišče »Jerneja« nima povezav. Tako dobimo predstavitev problema z grafom.

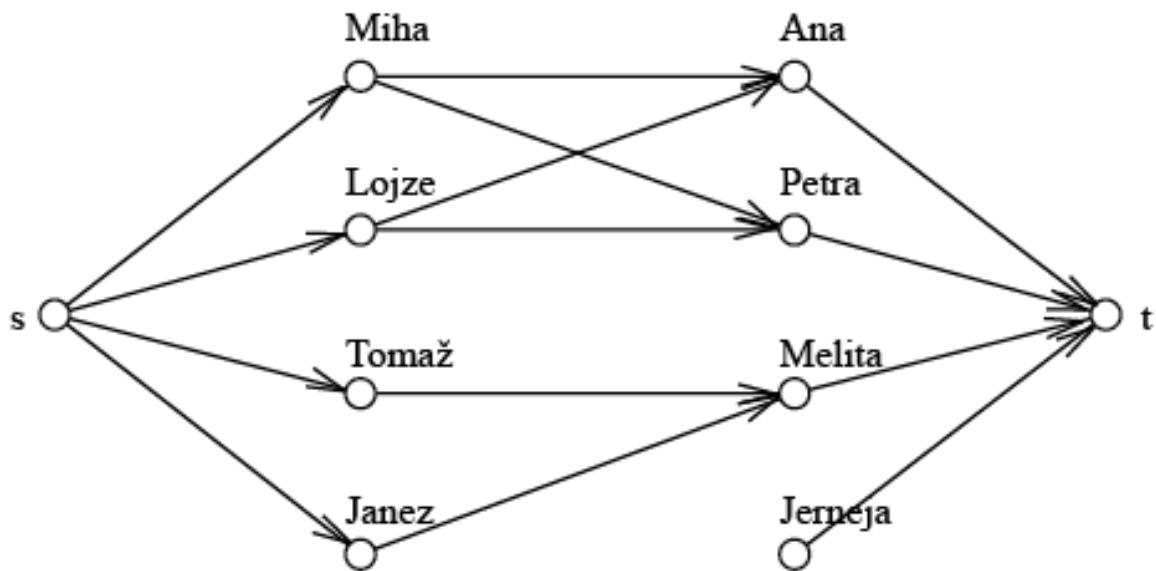
Lahko damo pa **usmerjene** (*Slika 2*) povezave, kjer imamo vozlišča: s (izvor) in t (ponor) ter ostala vozlišča, ki so imena vsake posamezne osebe v tabeli. Tu se bolj pregledno vidi kakšne kombinacije lahko izberem za poenostavljenou trdno zakonsko zvezo od s do t, ker je graf bolj povezan in ima pot.

Slika 1



*opomba Stolpci [Miha, Lojze, Tomaž, Janez] | Vrstice [Ana, Petra, Melita, Jerneja]. Stolpec Miha | Vrstica Melita.
0 | 1 → Miha ima rad Melito 1. Melita nima rada Mihe 0

Slika 2



Tu izbiramo posamezne povezave. V kolikor je usmerjena puščica iz enega vozlišča v drugega, obstaja možnost za trdno zakonsko zvezo, ki je obojestranska (drug drugega imata rada). Začnemo po vrsti vedno v vozlišču $s \rightarrow \text{Miha} \rightarrow \text{Ana} \rightarrow t$. Začnemo ponovno: $s \rightarrow \text{Lojze} \rightarrow \text{Petra} \rightarrow t$ (Ano smo že obiskali in je že v našem 1. seznamu kombinacij). Nadalujemo: $s \rightarrow \text{Tomaž} \rightarrow \text{Melita} \rightarrow t$. Od tu gremo od $s \rightarrow \text{Janez} \rightarrow t$ se ustavi, ker je Melita že obiskana. Dobili smo 1. možno kombinacijo. Tako nadajujemo postopek iskanja.

5.2.3

Za vsako osebo pogledamo seznam možnih partnerjev in, če ima možni partner v seznamu to osebo naredimo **neusmerjeno** (Slika 1) ali **usmerjeno** (Slika 2) povezano. Na koncu za iskanje vseh možnih parov za vsako vozlišče (osebe) pogledamo sosede. Dodamo posamezen par v seznam. Ko se premaknemo na naslednjo osebo pogledamo, če ni že vsebovana v seznamu parov. Če imamo veliko trdnih zakonov to pomeni več povezav in tudi več kombinacij. Graf lahko postane zelo nepregleden.



5.2.4 †

Partnerji	Miha	Lojze	Tomaž	Janez
Ana	1 \ 1 ^{topomba}	2 \ 2	3 \ ∞	∞ \ 1
Petra	2 \ 2	1 \ 1	∞ \ ∞	∞ \ ∞
Melita	∞ \ 3	∞ \ ∞	1 \ 1	2 \ 2
Jerneja	∞ \ 4	2 \ ∞	1 \ ∞	∞ \ 3

V tem primeru bi po prioriteti dobili trdne zakonske zveze. Seznam trdnih zakonskih zvez bi bil: $[[\text{Miha}, \text{Ana}], [\text{Lojze}, \text{Petra}], [\text{Tomaž}, \text{Melita}]] \rightarrow$ vse te osebe imajo med seboj najvišjo prioriteto 1.

^{topomba} Stolpci [Miha, Lojze, Tomaž, Janez] | Vrstice [Ana, Petra, Melita, Jerneja].

Stolpec Miha | Vrstica Petra. $2 \mid 2 \rightarrow$ Miha ima rad Petro s prioriteto 2. Petra ima rada Miha s prioriteto 2.

Stolpec Miha | Vrstica Ana. $1 \mid 1 \rightarrow$ Miha ima rad Ano s prioriteto 1. Ana ima rada Miha s prioriteto 1.

V tem primeru prioriteta 1 pomeni najvišjo prioriteto. ∞ pomeni, da oseba ni možen partner neke druge osebe.

V tabelo bi dodal prioriteto za vsak par. Nižja skupna vsota številk znotraj nekega polja bi pomenila višjo prioriteto. Za vsak par seštejemo vrednosti polja. Tisti stolpci z najnižjo skupno vrednostjo prikazujejo trdno zakonsko zvezo. Če bi imel en par **enako** {skupno} prioriteto (*Miha 1—2 Ana {3}, Miha 2—1 Petra {3}, Lojze 2—1 Ana {3}, Lojze 1—2 Petra {3}*) kot drug par **in** če bi znotraj tega para bili **različni** prioriteti 1—2 ali 2—1, potem te zveze **ne** bi bile trdne, ker bi prekršili **osnovno pravilo trdne zakonske zvezne** in bi se vsak partner oziroma partnerica imel vzajemno raje s partnerico oziroma s partnerjem drugega para, kot se imata partnerja v trenutni poroki in bi tako bilo narobe:

Lojze => Petra => Miha => Ana => Lojze



5.2.1 Za naslednje osebe imamo podane možne partnerje:

- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

Vsi možni trdni zakoni:

Miha-Ana, Miha-Petra, Lojze-Ana, Lojze-Petra, Tomaž-Melita, Janez-Melita

Ena izmed možnih rešitev:

Miha - Ana

Lojze - Petra

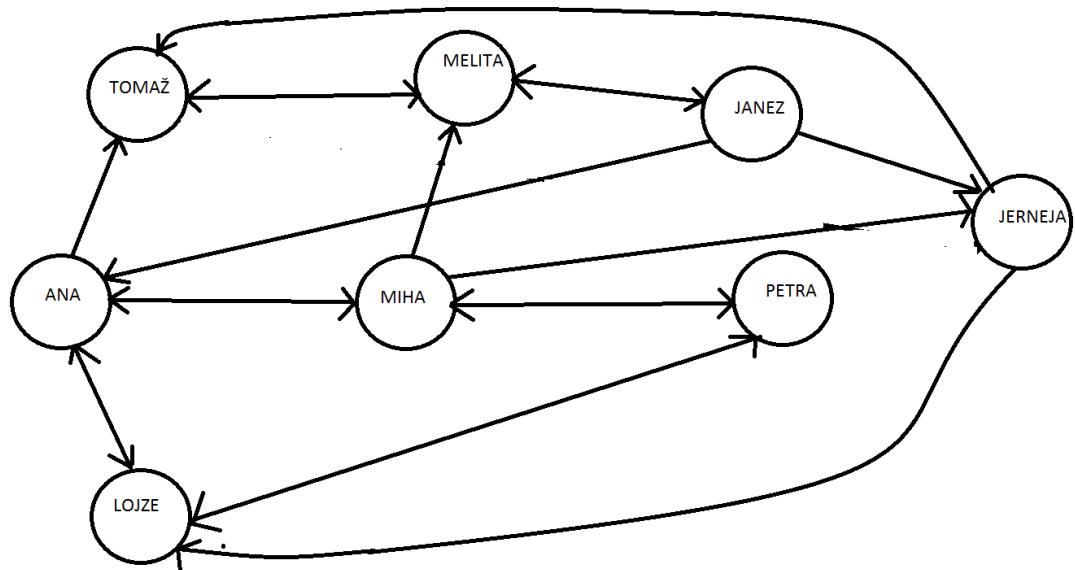
Tomaž - Melita

Janez in Jerneja v tem primeru ostaneta sama, saj so vsi njuni možni partnerji že v trdni zvezi, onedva pa nimata več možnih trdnih zakonov.

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

Vozlišča predstavljajo posamezne osebe, povezave pa predstavljajo možne partnerje neke osebe.

Glede na definicijo trdnega zakona, so smiselne samo povezave od neke osebe do oseb, z katerimi bi želela naša oseba biti v zvezi(torej povezav, ki ne predstavljajo možnih partnerjev ne rišemo).



Slika 1: Problem predstavljen z grafom

5.2.3 Opišite algoritem, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru? (Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

Algoritem začne v naključnem vozlišču ter preišče možne partnerje te osebe, dokler ne najde osebe, katera ima prav tako prvo osebo za partnerja(torej je med njima dvosmerna povezava). Nato ti dve osebi združi v par ter ju odstrani iz grafa in izpiše. Če za določeno osebo ne najde nobene take osebe(torej nima nobenih dvosmernih povezav - torej zanjo ne obstaja trdni zakon) algoritem preprosto nadaljuje z iskanjem trdih zakonov pri naslednji osebi.

Algoritem se konča, ko v grafu ni več nobenih dvosmernih povezav, oz. možnih trdnih zakonov.



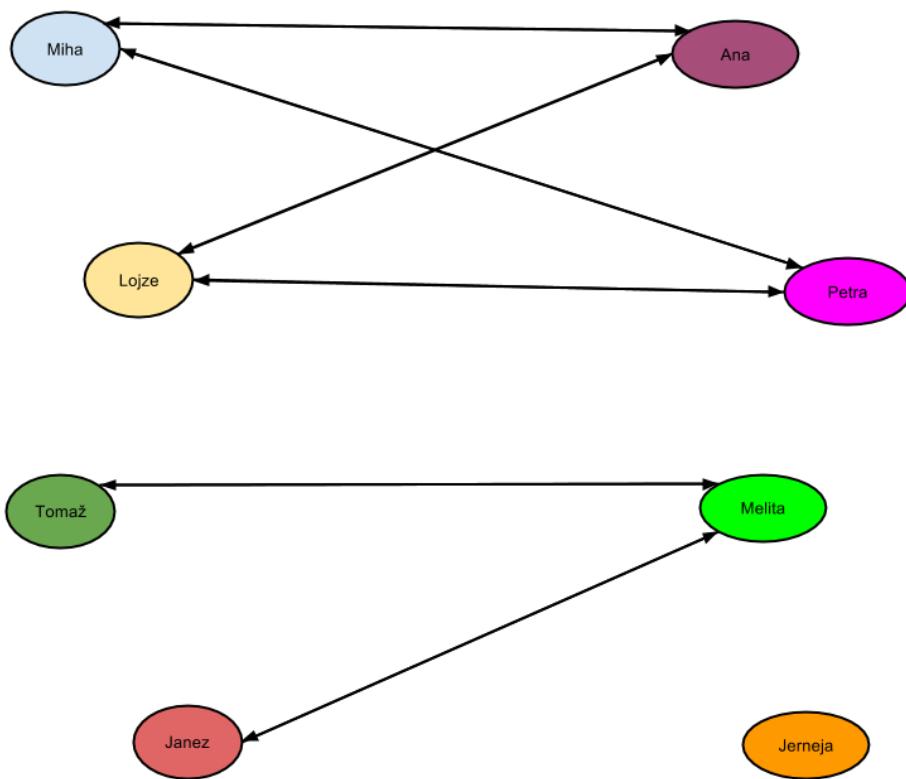
Naloga 5.2

5.2.1

Rešitev:

Miha – Ana
Lojze – Petra
Tomaž - Melita

5.2.2



5.2.3

Začnemo s tisto osebo, ki ima najmanj trdnih povezav. Če oseba ima 2 ali več trdnih povezav se odločamo glede na naključju ali pa prioriteti. V tem primeru se to zgodi pri Meliti ko mora izbrati ali Tomaža ali Janeza. Algoritem izbere enega. Če imamo veliko trdnih povezav pomeni da se algoritem mora na nek način odločati katero izbiro bo izbral. Ali bo to na podlagi koliko število povezav ima oseba B, ali glede na prioriteto.

Naloga 5.2.

5.2.1

	Miha	Lojze	Tomaž	Janez
Ana	1	1	1	1
Petra	1	1		
Melita	1		1	1
Jerneja	1	1	1	1

Možne rešitve so: M+A, L+P, M+T

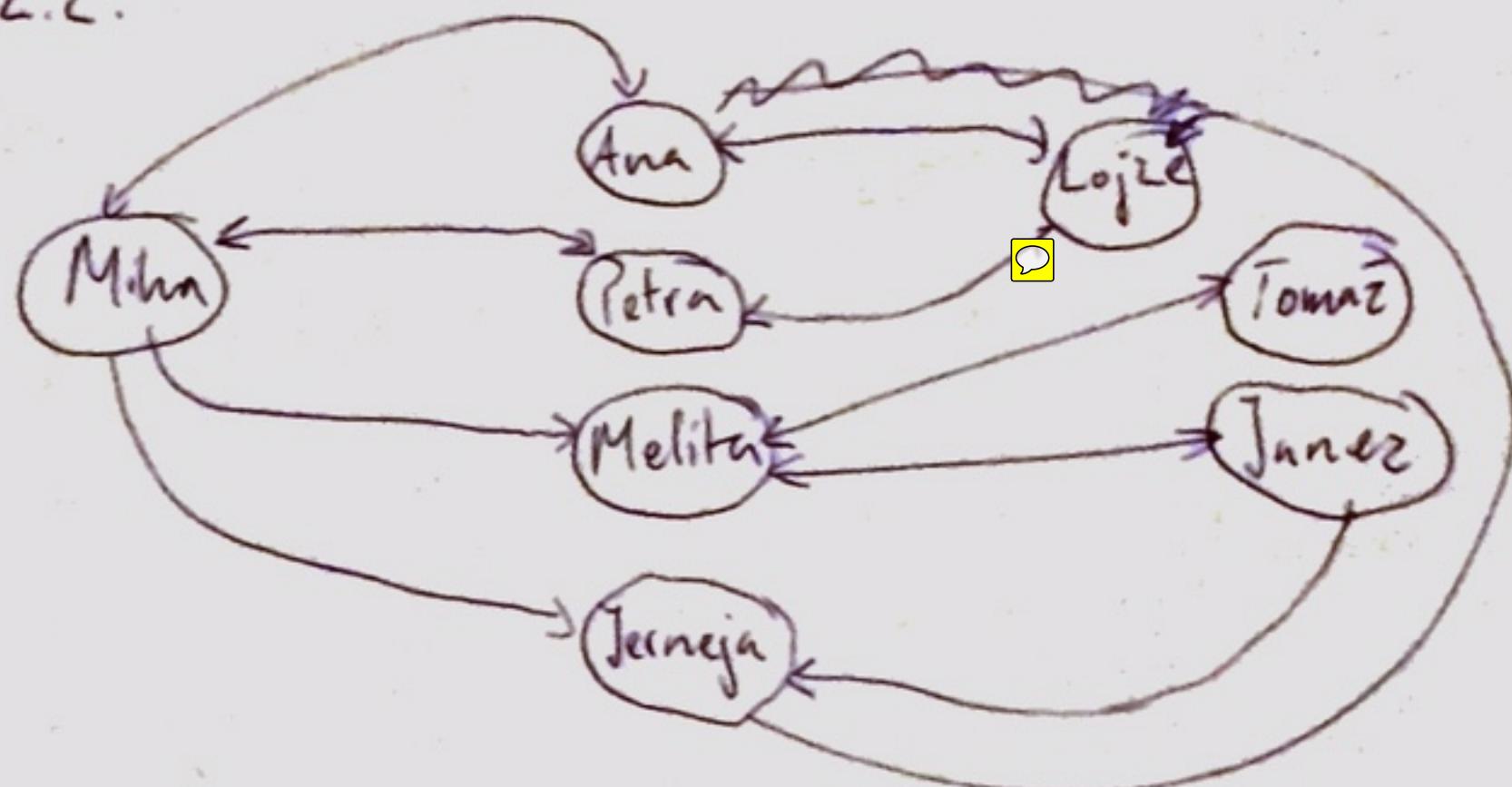
M+P, L+A, M+J

~~A+M, P+L, J+A~~

M+A, L+P, J+M

M+P, L+A, J+M

5.2.2.



Graf konstruiramo tako da so vozlišča osebe,

usmerjene povezave pa predstavljajo vero.

Obejstranska povezava pomeni trdni zakon

5.2.4

	Miha	Lojze	Tomaž	Janez
Ana	1.	2.	3.	
Petra	2.	1.	/	/
Melita	/	/	1.	2.
Jerneja	4.	3.	/	

Ravno tako b;

naredil za

moško fabelo, kjer

je prioriteta od 1. do

4. k je najnajvičen

5.2

5.2.1.

Lahko imamo 3 trdne zakone, našel sem 4 možnosti razporeditve parov:

Miha-Petra
Lojze-Ana
Janez-Melita

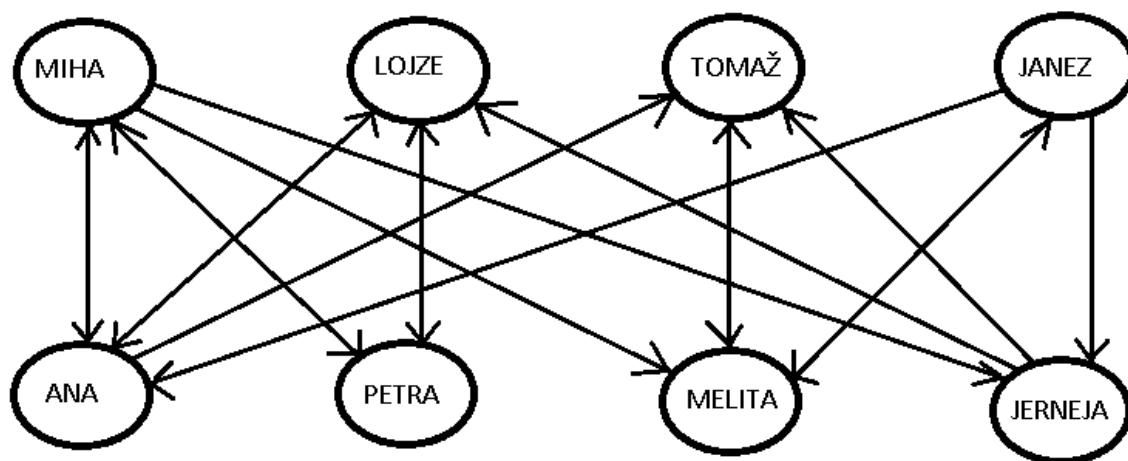
Miha-Petra
Lojze-Ana
Tomaž-Melita

Miha-Ana
Lojze-Petra
Janez-Melita

Miha-Ana
Lojze-Petra
Tomaž-Melita

5.2.2.

Vozlišče je oseba, povezava pa s kom bi se ta oseba rada poročila.



Problem bi z grafom rešili tako da bi poiskali dvosmerne povezave (tiste ki niso usmerjene samo v eno vozlišče), ko najdemo eno povezavo moramo osebi (vozlišči) odstraniti iz grafa, saj ima lahko ena oseba ustrezne povezave z več osebami, poročena pa je lahko samo z eno.

Različne kombinacije dobimo tako, da začnemo iskati na različnih koncih grafa.

5.2.3.

To, da imamo veliko trdnih zakonov pomeni, da je naš graf močno povezan.



Domača naloga 5

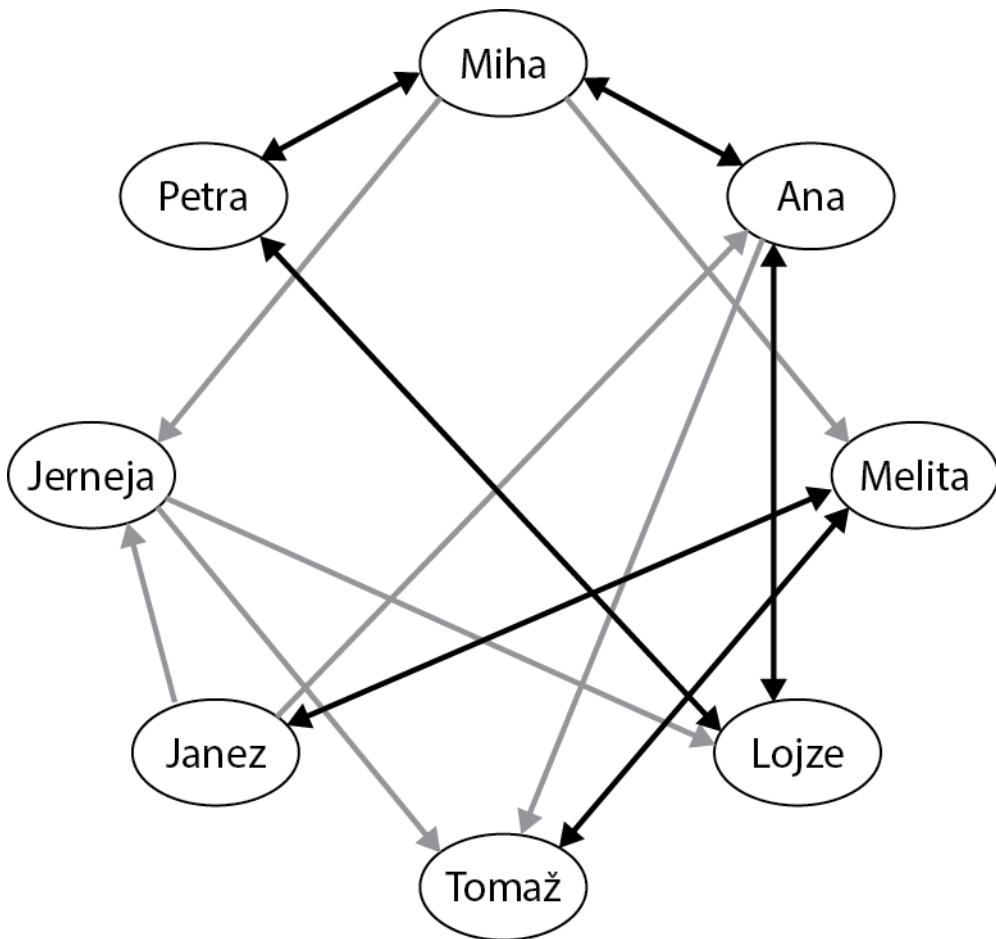
Naloga 2.1

Ena od možnih rešitev je: Lojze-Petra, Miha-Ana, Tomaž-Melita, Janez-Jerneja (ni trden).

Naloga 2.2

Vozlišča so posamezne osebe, povezave pa so možni partnerji (če je Miha izbral Ano, je povezava usmerjena od Mihe do Ane). Nekatere povezave so obojesmerne, kar pomeni, da sta partnerja izbrala en drugega. Te povezave je nujno predstaviti v grafu, ostale pa niso potrebne, saj za trden zakon potrebujemo obojestransko povezavo.

Graf za primer iz 5.2.1 izgleda takole:



Črne povezave (obojesmerne) so nujne, sive pa niso.

Naloga 5.2.3

Če imamo veliko trdnih zakonov (toliko, da se nam pojavijo cikli), moramo najprej odstraniti cikle. Ko to storimo, uredimo vozlišča po stopnjah. Potem vzamemo vozlišče A z najnižjo stopnjo (to je stopnja 1) in ga povežemo z edinim sosednjim vozliščem B. Nato izbrišemo vse povezave med vozliščem B in njegovimi sosednjimi vozlišči (razen povezave med B in A). Vzamemo naslednje vozlišče iz našega urejenega seznama vozlišč po stopnjah in ponovimo postopek. Število povezav, ki jih dobimo, ko gremo skozi vsa vozlišča naj bi bilo najmanjše možno število povezav.

Internetni viri mi pravijo, da je problem odstranjevanja ciklov NP težek problem.

APS NALOGA 5.2

5.2.1)

1. možna rešitev:

Miha, Ana
Lojze, Petra
Tomaž Melita

2. možna rešitev:

Janez, Melita
Lojze, Ana
Miha, Petra

3. možna rešitev:

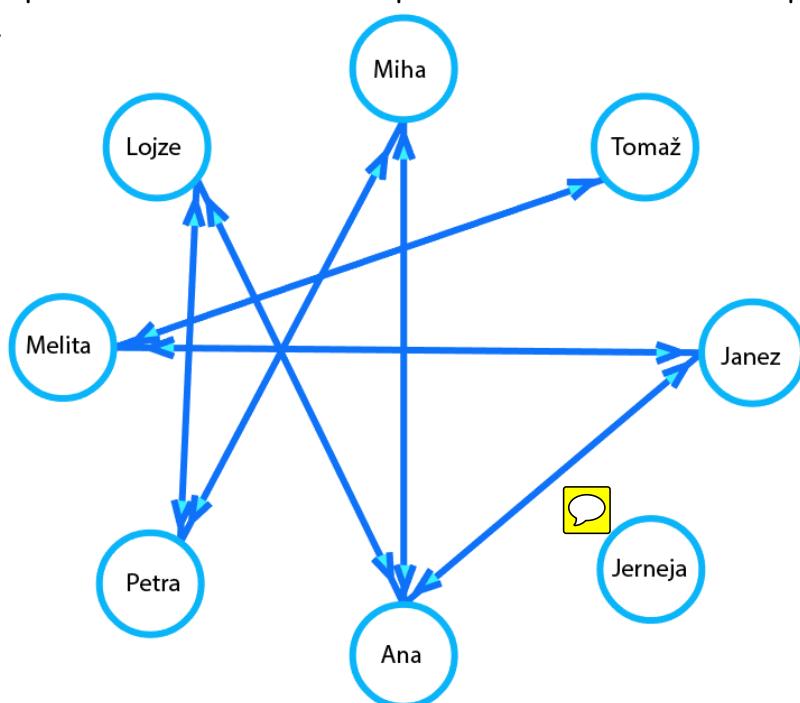
Melita, Tomaž
Ana, Lojze
Miha, Petra

4. možna rešitev:

Janez, Melita
Ana, Miha
Lojze, Petra

5.2.2)

Problem bi predstavil z grafom tako, da bi ogovor napisal imena ljudi, povezave pa bi bile relacije med njimi. Obojestranska povezava bi pomenila trdne povezave in le obojestranske povezave so smiselne za ta problem tako da enosmernih povezav sploh ne potrebujemo.



5.2.3)

```
stableMatching()
{
    M=množica fantov
    W=množica deklet
    Dokler obstaja m v M ki še ni poročen:
    {
        w = prvo dekle v W ki je všeč m
        če je w prosta:
            (m, w) se poročita. 
        drugače:
            če ima w rajši m kot m':
                (m, w) se poročita.
                m' postane prost.
            drugače:
                (m', w) ostaneta poročena.
    }
}
```

APS 5. domača naloga

Naloga 5.2

Problem trdne zakonske zveze. V matematiki in računalništvu je znan problem trdne zakonske zveze (angl. *stable marriage*). Pri tem problemu imata vsak fant in vsako dekle preferenco, s kom bi se rad oziroma rada poročila ter želimo poroke določiti tako, da so trdne. Poroka med fantom F in dekletom D je trdna, če hkrati ne velja:

- da obstaja dekle D', ki bi jo fant F imel raje kot dekle D; in
- da ne obstaja fant F', ki bi ga imelo dekle D raje kot fanta F.

Z drugimi besedami, želimo dobiti takšne pare, da ne obstajata nobeno dekle in fant, za katera bi veljalo, da bi se imela vzajemno raje, kot pa imata rada partnerja v trenutni poroki.

Za obvezni del naše naloge bomo problem nekoliko *poenostavili*. Tako bo preferenca zgolj binarno število 1 - se želi poročiti z osebo, ali 0 - se ne želi poročiti z osebo. Zakon velja za trdnega, če se tako F želi poročiti z D kot tudi obratno. Naloga je, da poiščete čim več trdnih zakonov.

Vprašanja:

5.2.1 Za naslednje osebe imamo podane možne partnerje:

- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

REŠITVE:

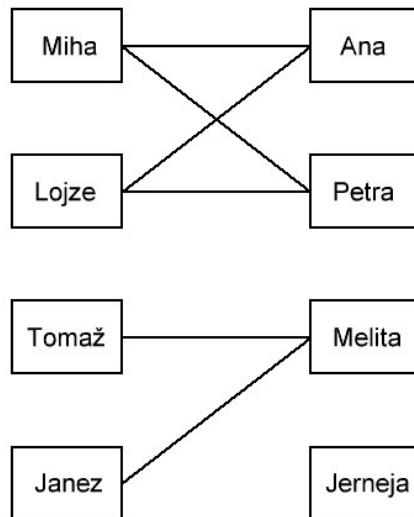
- Miha - Ana, Lojze - Petra, Janez - Melita
- Miha - Petra, Lojze - Ana, Janez - Melita
- Miha - Ana, Lojze - Petra, Tomaž - Melita
- Miha - Petra, Lojze - Ana, Tomaž - Melita

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

REŠITEV:

Vozlišče predstavlja ime določene osebe. Na eni strani imamo fante (x), na drugi dekleta(y). Povezava nam pove, če se določena oseba želi poročiti z drugo osebo.

(x) (y)



5.2.3 Opišite algoritem, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru? (Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

REŠITEV:

Opis algoritma: imamo seznam moških in seznam žensk. Vsaka oseba imajo svoj seznam oseb, ki jih želi.

Algoritem:

For moski v seznamuMoskih do:

for zenska v seznamuZensk do:

if zenska v seznamu zelja moskega in moski v seznamu zelja zenske then:

naredi povezavo od moskega do zenske



5.2.4 (Dodatna) Končno rešujemo še splošni problem trdne zakonske zveze. V tem primeru preference predstavljajo rangiranje partnerjev glede na željo po poroki posamezne osebe. Predlagajte rešitev in utemeljite njeno pravilnost. (Namig: V resnici pri reševanju splošnega problema ne uporabimo modeliranja z grafom.)

naloga 5.25.2.1

	Miha	Lojze	Tomaž	Janez
Ana	1	1	0	1
Petra	1	1	0	0
Melita	0	0	1	1
Jerneja	0	1	1	0

Trdni zakoni:

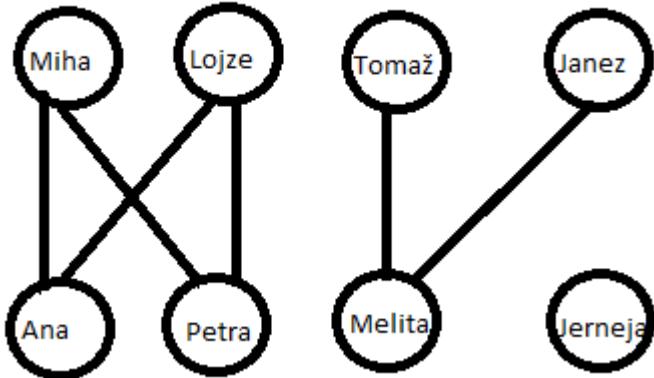
1. Miha + Ana, Lojze + Petra, Tomaž + Melita;
2. Miha + Ana, Lojze + Petra, Janez + Melita;
3. Miha + Petra, Lojze + Ana, Tomaž + Melita;
4. Miha + Petra, Lojze + Ana, Janez + Melita;

Netrdni zakon: (v primeru, da se morajo poročiti vsi posamezniki)

1. Janez + Jerneja
2. Tomaž + Jerneja
3. Janez + Jerneja
4. Tomaž + Jerneja

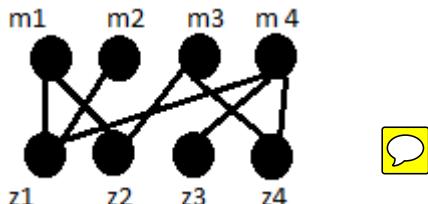
5.2.2

Točke bi predstavljali vsi posamezniki, smiselna povezava pa bi bila trdna zveza med partnerjema. Tako najdemo trdne zveze samo s povezavami med vozlišči.



5.2.3

Vzamemo prvi člen v grafi in pogledamo, če ima kakšno povezavo. Če ima samo eno mu določimo to zvezo, drugače pogledamo po prvi povezavi, če ima še kakšnega izbranca. Če ga nima poročimo ta par, če pa ima pa pogledamo ali je morda bolj smiselno poročiti drug par in določiti naslednjo osebo prvemu členu. To ponovimo za vse preostale člene. Primer:



Pogledamo člen m1. Prva trdna povezava je z z1, torej sledimo tej povezavi. Z1 ima trdno povezavo še z m2 in m4, pogledamo ali imata druge izbire. Izkaže se da je najbolj smiselno poročiti z1 in m2, ker ima m2 samo 1 trdno povezavo. Vrnemo se na m1, ki ima samo še eno trdno povezavo z z2, torej ga poročimo z z2. Skočimo na m3, ker je m2 že poročen. M3 ima prvo trdno povezavo z z2, ki je že zasedena, torej mu ostane samo z4. Pogledamo še zadnji člen m4. Ta ima tri trdne povezave vendar je samo ena prosta tako da mu dodelimo z3.

5.2.4

Za primer si izberemo dopolnjen in malo spremenjen seznam iz prve naloge:

Miha: Jerneja, Ana, Petra, Melita

Lojze: Ana, Petra, Jerneja, Melita

Tomaž: Jerneja, melita, Petra, Ana

Janez: Ana, Melita, Jerneja, Petra

Ana: Miha, Lojze, Tomaž, Janez

Petra: Lojze, Miha, Janez, Tomaž

Melita: Tomaž, Janez, Lojze, Miha

Jerneja: Tomaž, Lojze, Miha, Janez

Najprej pogledamo prve želje mož. Prva Mihova želja je Jerneja. Ta je na njenem seznamu bolj pri dnu, vendar, ker je še ni nihče zaprosil, ga ne zavrne.

Naslednji na vrsti je Lojze. Lojze zaprosi Ano, ta sprejme ker, je še nihče ni zaprosil. Sledi Tomaž, ki zaprosi Jernejo. Ta je že sprejela prošnjo od Mihe,

vendar je Tomaž njena prva želja tako da Miho zavrne in sprejme Tomaža.

Zadnji na vrsti je Janez, ki zaprosi Ano, vendar je ta že zaročena z Lojzetom, ki je višje na seznamu, zato ga zavrne. Ponovno gremo od začetka. Mihova naslednja želja je Ana. Ta je že zaročena z Lojzetom, vendar je Miha njena prva želja zato se zaroči z Mihom. S tem dobimo že 2 trdna para Miha + Ana in Tomaž + Jerneja. Naslednji na seznamu je Lojze, ki je ravno izgubil nevesto. Zaprosi Petro, ta sprejme ker je Lojze njena prva želja. Tako ostaneta samo še Janez in Melita za zadnjo trdno zvezo.

Naloga 5.2

5.2.1

Vsaj ena možna rešitev je

Miha – Ana

Lojze – Petra

Tomaž – Melita

Janez

Jerneja

5.2.2

Vozlišče je oseba. Povezava je poroka. Le trde poroke so smiselne povezave.



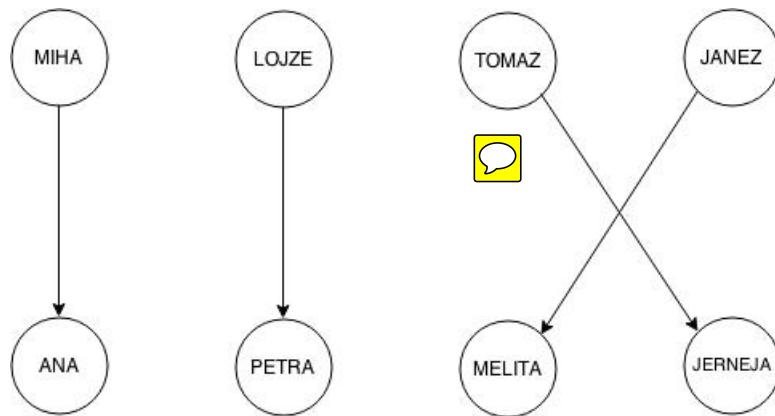
5.2.3

1

Ena izmed možnih rešitev:

- Janez & Melita
- Tomaž & Jerneja 
- Miha & Ana
- Lojze & Petra

2



Vozlišča predstavljajo osebe, povezave pa določajo trdne zakone (binarno število je enako 1). Iz enega vozlišča lahko naredimo največ 1 povezavo.

3

4

Uporabimo Gale-Shapelyjev algoritmom:

```
def stableMarriageMatch(Proposers, Proposed):  
    // Vrnemo na koncu  
    Matched = []  
  
    Unmatched = list(Proposers.keys())  
    UnmatchedCopy = Unmatched  
  
    // Preslikava iz seznama fantov v indeks naslednjega dekleta v vrsti za poroko  
    ListOfPreviousMatched = dict(zip(Unmatched, [0]*len(Proposers)))  
  
    while (len(UnmatchedCopy) > 0):  
        for (U in UnmatchedCopy):  
            Upartner = Proposers[U][ListOfPreviousMatched[U]]  
  
            if (Upartner in Matched):  
                if (Proposed[Upartner].index(U) < Proposed[Upartner].index(Matched[Upartner])):  
                    Unmatched.append(Matched[Upartner])  
                    Unmatched.remove(U)  
                    Matched[Upartner] = U  
            else:  
                pass
```

```
    Matched[UPartner] = U
    Unmatched.remove(U)

    ListOfPreviousMatched[U] = ListOfPreviousMatched[U] + 1
    UnmatchCopy = Unmatched
    return Matched
```

- Algoritem se vedno zaključi.
- Na koncu izvajanja vsak fant dobi optimalno optimalno poroko.
- Na koncu izvajanja vsako dekle dobi najmanj optimalno poroko.
- Na koncu ne moreta obstajati neporočena fant in dekle, saj jo je moral v nekem trenutku zaprositi (fant bo zaprosil vsa dekleta, če je to potrebno), ona pa je potem zagotovo z nekom zaročena.

<http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15251-f10/Site/Materials/Lectures/Lecture21/lecture21.pdf>

Naloga 5.2

5.2.1 ena možna rešitev

Nekaj možnih rešitev je:

Tomaž – Melita

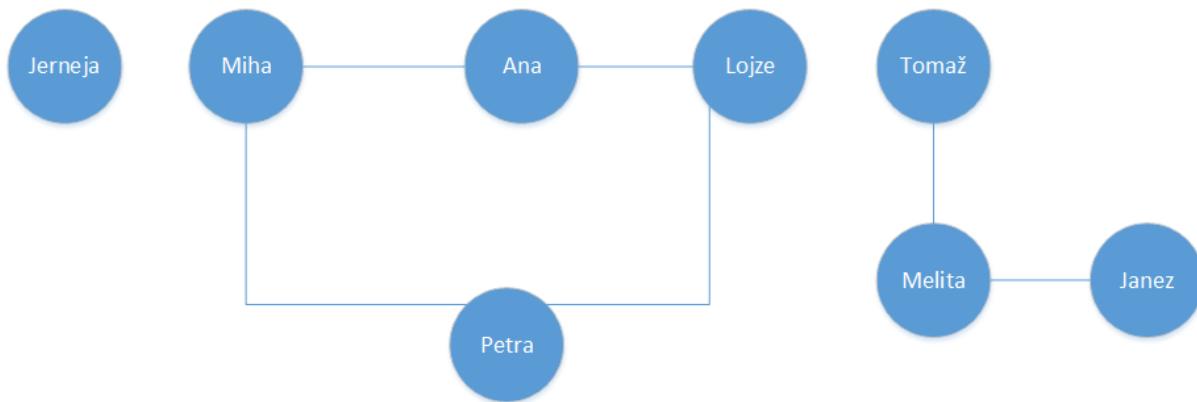
Lojze – Petra

Miha – Ana

5.2.2 predstavitev trdnega zakona z grafom

Ker je trden graf definiran s tem, da se prva oseba želi poročiti z drugo in obratno, je graf lahko neusmerjen. Povezave pa so tam, kjer to velja za oba. Torej, vozlišče je oseba, povezava pa je, če je med njima trdna zveza.

Spodnji graf prikazuje možne poroke. Prikazuje vse trdne povezave v grafu. Graf sicer ni povezan med samo, ampak razdeljen na več skupin.



5.2.3 Opis algoritma

Sestavljanje grafa:

Recimo, da podatke prejmemo tako, da imamo en seznam, v katerem imamo shranjeno kot na primeru – torej (Oseba, seznam možnih partnerjev). Tako za vsako osebo vidimo hitro, kateri so njegovi možni partnerji.

Gremo skozi celoten dobljen seznam, in sicer:

- Najprej se sprehodimo skozi vse **Osebe** v prejetem seznamu, ter za vsako osebo naredimo svoje vozlišče.
- Ko imamo vsa vozlišča, vseh oseb, se lotimo povezav. Zopet gremo skozi celoten seznam. Najprej vzamemo osebo, ter pogledamo *seznam njenih možnih partnerjev* (v nadaljevanju *sezMP*).
- Vzamemo prvo osebo iz *sezMP*, ter naredimo do nje **usmerjeno** povezavo. Nato vzamemo drugo osebo in zopet naredimo **usmerjeno** povezavo. Tako naredimo za vse osebe v *sezMP*.
- Ko pridemo do konca *sezMP*, ponovimo drugo ter tretjo alinejo, s tem da kot osebo vzamemo naslednjo. To ponavljamo dokler nam ne zmanjka oseb (dokler ne napišemo povezav tudi za zadnjo osebo v prvotnem prejetem seznamu)

- Ko imamo napravljene vse (usmerjene) povezave, jih moramo še združiti v neusmerjene, ostale pa izbrisati. To storimo tako, da se sprehodimo čez celoten seznam oseb, ter za vsako osebo njegovih povezav (*sezMP*) in gledamo: Če je povezava od osebe A do osebe B, ter povezava od osebe B do osebe A, potem to povezavo združimo v neusmerjeno. (Ubistvu, v kodi lahko tudi pustimo obe povezavi, ne bo nič narobe). Če pa povezava od osebe A do osebe B obstaja, vendar od osebe B do osebe A ne, potem to povezavo izbrišemo, saj tak zakon nebi bil trden.

Iskanje trdnih zakonov:

Sedaj, ko imamo predstavljen graf kot je zgoraj, lahko trdne zakone najdemo tako, da se le sprehodimo čez vse obstoječe povezave. Ker vsaka povezava pomeni trdno zvezo, potem samo izpišemo obe osebi. To bi sicer našlo (če vzamemo graf iz 5.2.2 naloge) Tomaž – Melita ter Melita – Janez. Našlo bi torej vse možne trne zakone. Seveda, če vzamemo situacijo, da Melita ne sme biti pri obeh, potem bi morali vozlišča ob iskanju označiti, da so že »zasedena«. S tem bi rešili ta problem.



5.2.1

Da sem nalogo hitreje rešil sem si pomagal s tabelo. Pregledal sem vse osebe in s tistimi, ki bi se ta oseba poročila sem označil z X. Tam kjer sta 2 X pride do trdnih zvez:

	MIHA	LOJZE	TOMAŽ	JANEZ
ANA	XX	XX	X	X
PETRA	XX	XX		
MELITA	X		XX	XX
JERNEJA	X	X	X	X

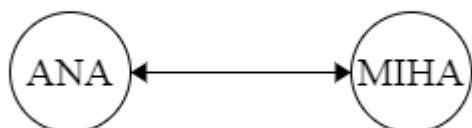
Do trdnih zvez pride med:

1. Miha in Ana
2. Miha in Petra
3. Lojze in Ana
4. Lojze in Petra
5. Tomaž in Melita
6. Janez in Melita



5.2.2

Problem bi rešil z povezavami ki so usmirjene v obe strani. Povezava pomeni trdnost zveze, vozlišče je pa oseba. Primer med Ano in Miho:



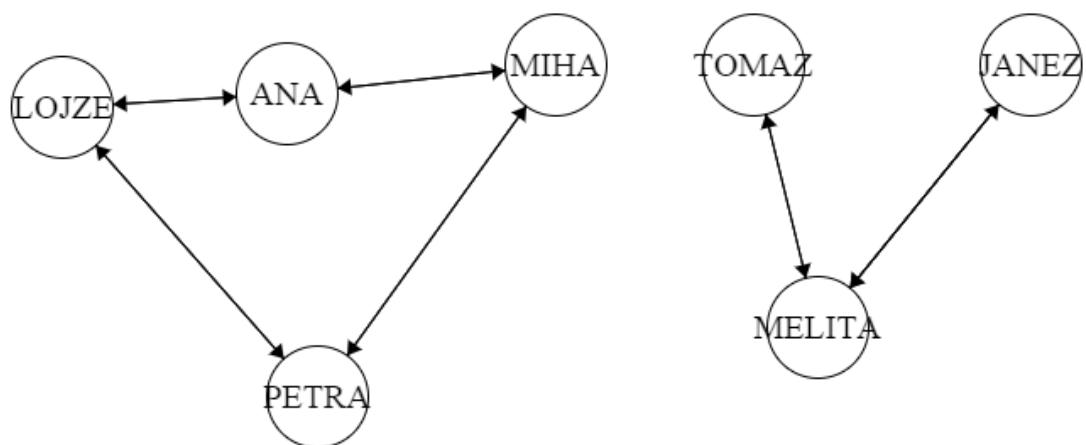
Ana bi se poročila z Miho – dobimo povezavo umserjeno iz Ane v Miho.

Miha bi se poročil z Ano – dobimo povezavo usmerjeno iz Mihe v Ano.

Tako dobimo povezave v obe smeri.

Ker gledamo le za trdne zakone, potrebujemo le povezave, ki so usmerjene v obe strani. Enostranske povezave odpadejo.

Graf za osebe iz prve naloge:



Vozlišča JERNEJA ne potrebujemo, ker nima z nobenim trdne zveze.

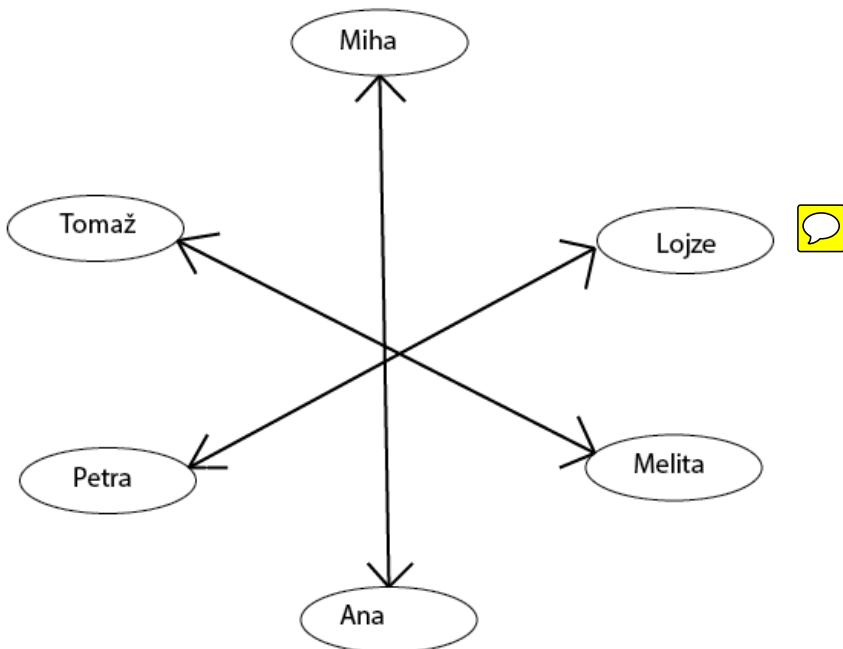
5.2.1

Ena rešitev je lahko lahko:

- Miha in Ana
- Petra in Lojze ter
- Tomaž in Melita

5.2.2

Vozlišča so osebe, povezave pa predstavljajo trdne zveze. Smiselne so samo trde.



V tej rešitvi manjkata še Jerneja in Janez toda njuna povezava ni dvosmerna(trdna), zato je nisem narisal.

5.2.3

Algoritem deluje tako, da preverimo ali so vozlišča še prosta, torej nima povezave. Če je prost pogledamo v seznam željenih »partnerjev«, in enemu pošljemo prošnjo za »poroko«- Če je fant v njenem seznamu(drugo vozlišče), se ta zaročita in naredi se trdna povezava. Ti vozlišči nista več na voljo drugim in se črtata iz seznamov ali pa preskakujeta odvisno od implementacije. V primeru da je dekle že zasedeno, fant ponavlja postopek in pošilja vsem puncam prošnjo dokler ne pride do prostega vozlišča ali pa do konca svojega seznama. Optimalno bi bilo da, bi vsako vozlišče imelo samo eno trdo povezavo, tako algoritem deluje najhitreje.



naloga 5.2

Problem trdne zakonske zveze. V matematiki in računalništvu je znan problem trdne zakonske zveze (angl. *stable marriage*). Pri tem problemu imata vsak fant in vsako dekle preferenco, s kom bi se rad oziroma rada poročila ter želimo poroke določiti tako, da so trdne. Poroka med fantom F in dekletom D je trdna, če **hkrati ne velja**:

- da obstaja dekle D', ki bi jo fant F imel raje kot dekle D; in
- da ne obstaja fant F', ki ga imelo dekle D raje kot fanta F.

Z drugimi besedami, želimo dobiti takšne pare, da ne obstajata nobeno dekle in fant, za katera bi veljalo, da bi se imela vzajemno raje, kot pa imata rada partnerja v trenutni poroki.

Za obvezni del naše naloge bomo problem nekoliko **poenostavili**. Tako bo preferenca zgolj binarno število 1 - se želi poročiti z osebo, ali 0 - se ne želi poročiti z osebo. Zakon velja za trdnega, če tako F želi poročiti z D kot tudi obratno. Naloga je, da poiščete čim več trdnih zakonov.

Vprašanja:

5.2.1 Za naslednje osebe imamo podane možne partnerje:

- Miha: Ana, Petra, Melita, Jerneja
- Lojze: Petra, Ana
- Tomaž: Melita
- Janez: Ana, Melita, Jerneja
- Ana: Miha, Lojze, Tomaž
- Petra: Lojze, Miha
- Melita: Tomaž, Janez
- Jerneja: Tomaž, Lojze

Kakšna je vsaj ena možna rešitev?

Tomaž – Melita

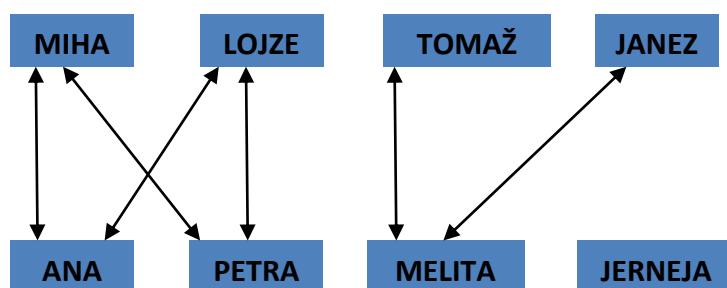
Petra – Lojze

Ana-Miha

Brez para ostaneta Jerneja ter Janeza, ker Jerneja nima pod partnerji Janeza.

5.2.2 Kako bi problem predstavili z grafom. (Namig: Kaj je vozlišče in kaj povezava. Katere povezave so sploh smiselne glede na definicijo trdnosti? Narišite graf za osebe podane v točki 5.2.1.).

Vozlišče bi bila oseba, povezava pa bi bila do tistega ki se omenja kot možni partner. Smiselne povezave so tiste ki imajo obojestransko povezavo se pravi od vozlišča A do B ter B do A. Druge povezave lahko odstranimo zaradi definicije trdnosti.



5.2.3 Opišite algoritem, ki reši poenostavljeni problem trdne zakonske zveze na splošnem primeru? (Namig: Uporabite predstavitev z grafom ter premisliti, kaj pomeni, da imamo veliko trdnih zakonov?)

Z algoritmom bi zgradil graf kot na zgornjem primeru.

Odstranil bi vsa vozlišča ki nimajo povezave saj nam nič ne koristijo ker ga ne moremo uvrstiti v nobeno trdno zakonsko zvezo.

Po vrsti bi potem določal zakonske zveze na naslednji princip.

Izberemo vozlišče z 1 povezavo ter ju določimo za par ter jih odstranimo z grafa.

Odstranimo vse povezave do odstranjenih vozlišč ter morebitna vozlišča ki so ostala brez povezav.

Ponavljamo dokler nismo odstranili vsa vozlišča ki imajo samo 1 povezavo, ter potem ponovno prvemu z 2ma vozliščema izberemo par. Ni pomembno katerega vzamemo.

Itd z več povezavami. Vedno vzamemo tistega z najmanj povezavami.

Tako bi izkoristila kar največ povezav v grafu, ter tako določili karseda veliko trdnih zakonskih zvez.

5.2.4 (Dodatna) Končno rešujemo še splošni problem trdne zakonske zveze. V tem primeru preference predstavljajo rangiranje partnerjev glede na željo po poroki posamezne osebe. Predlagajte rešitev in utemeljite njeno pravilnost. (Namig: V resnic pri reševanju splošnega problema ne uporabimo modeliranja z grafom.)

1)

V našem primeru (torej poenostavljena trdna zakonska zveza) dobimo kar cel kup parov;

Miha – Ana

Miha – Petra

Lojze – Ana

Lojze – Petra

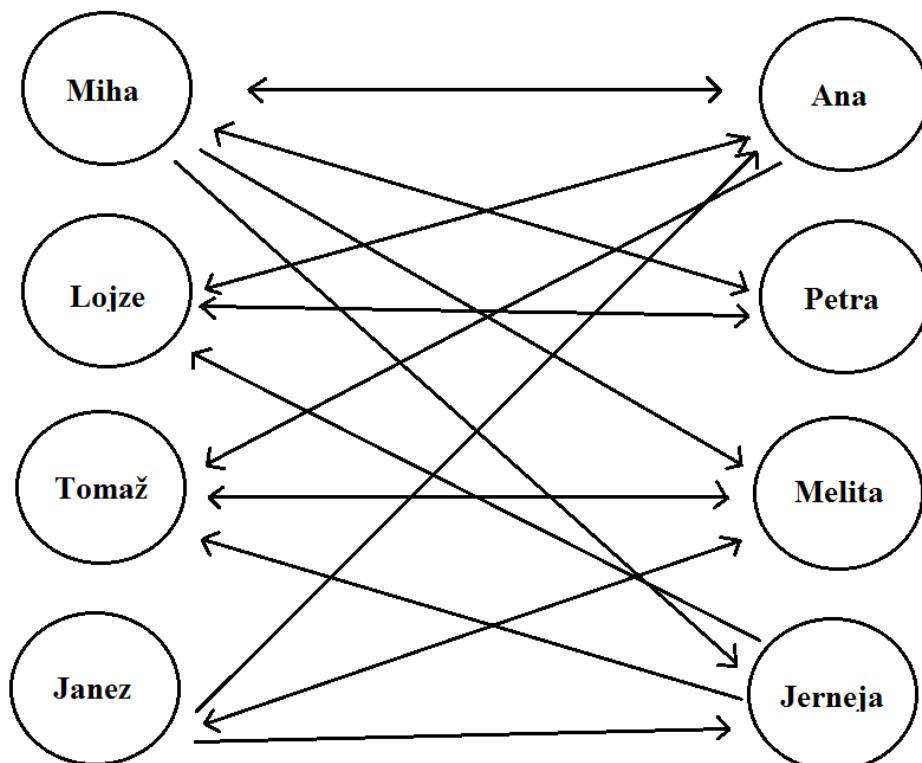


Tomaž – Melita

Janez – Melita

2)

Primer v nalogi 1) prikazan z grafom;



Vozlišča so tu vse osebe, povezave pa ljubezenski interesi (puščica proti osebi nakazuje interes osebe od koder povezava poteka). Povezava, kjer sta na obeh straneh puščici je trdna povezava (oba se želita poročiti eden z drugim).

4)

Algoritem za reševanje problema splošne trdne zakonske zveze je sila preprost;

- Vsak posameznik rankira vse osebe nasprotnega spola po preferenci
 - o Vsak moški »zaproši« prvo osebo na njegovi lestvici
 - o Vsaka ženska »zavrne« vse razen najvišjega ocenjenega moškega na njeni lestvici, ki jo je zaprosil
 - o Vsi moški, ki so bili zavrnjeni v prejšnjem koraku spet zaprosijo prvo osebo na njihovi lestvici (izvzemajoč osebo, ki so jo prej zaprosili)
 - o Vsaka ženska spet zavrne vse razen najvišje ocenjenega moškega na njeni lestvici, ki jo je zaprosil
 - o Ti koraki (zaprošitve, zavrnitve) se ponavljajo, dokler niso vse osebe zaročene.

Miha – Ana

Lojze – Petra

Tomaž – Melita

Pri vseh treh zgornjih parih so odgovori trivialni, saj se imata pri vseh partnerja vzajemno najraje glede na podane preference.

Janez – Ana (pri Ani je sicer Miha na prvem mestu, a je pri Janezu Ana – kar je dovolj)

Jerneja – Tomaž (pri Tomažu je Melita na prvem mestu, a je pri Jerneji Tomaž, kar je spet dovolj)



a) Zapišite vrstni red izpadanja za $m = 7$, $n = 30$.

Pozor: Upoštevati morate, da otrok, ki so izpadli, ne upoštevamo več.

Vrstni red ispadanja:

7,14,21,28,5,13,22,30,9,18,27,8,19,1,12,25,19,24,11,29,17,6,3,2,4,16,26,15,20,23