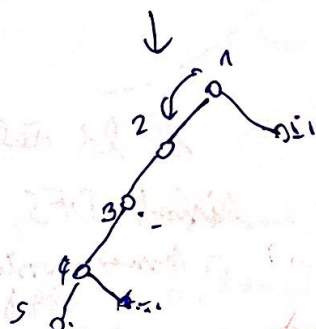
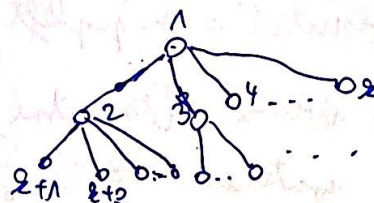


6. VAJE

NALOGA 1 : Izvedi BFS algoritem.

BFS = Breadth first search

DFS = Depth first search



BFS algoritem:

- uporabi se za pregled grafa
- vrsto dveh vrst v grafu, povezane komponente
- za preverjanje dolžnosti grafa
- iskanje najkrajših poti v ~~re~~ (neusmerenem grafu)

↓ implementacija

$$\text{def BFS}(G, n):$$

Realizace

graf Et zezum  
zeedov

$$n = \text{len}(G)$$

obiskani = [False] \* n

$Z = \text{sort}([w])$  # řadíme ~  $w$   
(DFS: slab(1))

~ python:

from collections import

*Deques*

while  $z$ :

trénutní = z. popřít()

if obisani [trenutni] :

continue # ms ya ite obiskali

obissani [trenutni] = True

for seed in G[nemutai]:

if not disbani [posed];

8. push (used)

is a simple ~~linear~~ <sup>linear</sup> tree

(samana hah is  
love ali is done)

re la zelele.

inst: DFS, run  
monitors write monitoring  
read

noch nur element

Ma nte

Append

22.2. Ali tenar algoritmu doboli že naslednje (d):

⊙ BFS, az izmij kot:

↓ na drugi stran

②



def BFS (G, u):

'''Vrne najkrajšo poti od u do vseh ostalih vozlišč'''

n = len(G)

d = [0] \* n

obiskani = [False] \* n

q = vrsta([(u, 0)])

while q: # dokler vrsta ni prazna

trenutni, razdalja = q.popleft()

if obiskani[trenutni]: continue

obiskani[trenutni] = True

d[trenutni] = razdalja

for sosed in G(trenutni):

if not obiskani[sosed]:

q.push((sosed, razdalja + 1))

return d

ČASOVNA ZAHTEVOST:  $O(n+m)$

n... št. vozlišč

m... št. povezav

Če bi želeli iskaniti najkrajšo pot, bi morali v vrsto dodati  
ne en parameter, ki predstavlja starša trenutnega vozlišča. Tako bi  
razponumeno in bi smo prišli do trenutnega vozlišča:

③

q = vrsta([(u, 0, u)])

## NALOGA 2: Floyd - Warshall algoritem

FW algoritem:

VHOD: graf  $G$ , utežen (negativne uteži dovoljene)

IZHOD:  $D$  dimenzije  $n \times n$  ( $n$  št. vozlišč)

(\*)  $D_{ij}$  ... cena najkrajše poti med  $i$ -tim in  $j$ -tima vozliščem

IDEJA:

$$D_{ij}(k) = \min \{ D_{ij}, D_{ik} + D_{kj} \} \quad \min \{ D_{ij}(k-1), D_{ik}(k-1) + D_{kj}(k-1) \}$$

↑  
isto št.  $D_{ij}^*$  samo,

da upoštevamo vozlišča

od 1 do  $k$

ROBNI POGOJI:

$$D_{ii}(1) = 0$$

$$D_{ii}(1) = w_{ii} \quad (\text{utež povezan})$$

ČASOVNA ZAHTEVNOST:

$$O(n^3)$$

stoper in potem ~~10~~ pomenuje poti.

~~7m, 7m, 7m~~ raje

### NALOGA 5:

Če bi,  $P$  FK oddali, če v grafu obstajajo negativni rožli?

Tukaj, da bi preverili diagonalne elemente v končni matriki.

Če je kateri od njih negativen, potem ima graf negativne rožle.