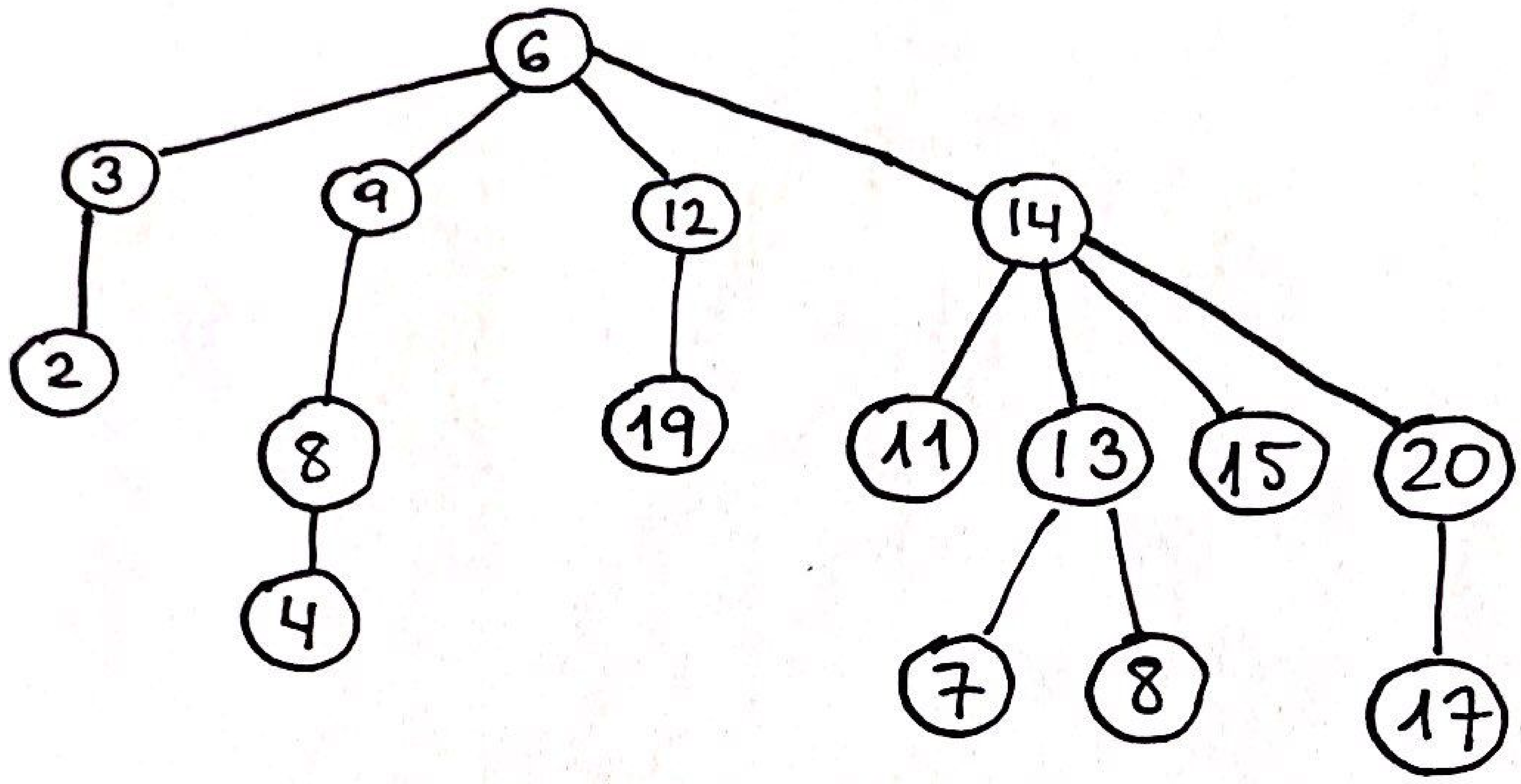
H1: BREITENSUCHE UND TIEFENSUCHE.

Iteration	u	v	Q
0	–	□	[6]
1	6	3,9,12,14	[3,9,12,14]
2	3	2	[9,12,14,2]
3	9	8	[12,14,2,8]
4	12	19	[14,2,8,19]
5	14	11,13,15,20	[2,8,19,11,13,15,20]
6	2	□	[8,19,11,13,15,20]
7	8	4	[19,11,13,15,20,4]
8	19	□	[11,13,15,20,4]
9	11	□	[13,15,20,4]
10	13	7,18	[15,20,4,7,18]
11	15	□	[20,4,7,18]
12	20	17	[4,7,18,17]
13	4	□	[7,18,17]
14	7	□	[18,17]
15	18	□	[17]
16	17	□	— (leer)

a) i

Knoten	3	9	12	14	2	8	19	11	13	15	20	4	17	18	7	1	10	5	16	6
Distanz	1	1	1	1	2	2	2	2	2	2	2	3	3	3	3	∞	∞	∞	∞	Ø
Vorg.	6	6	6	6	3	9	12	14	14	14	14	8	20	13	13	NIL	NIL	NIL	NIL	NIL

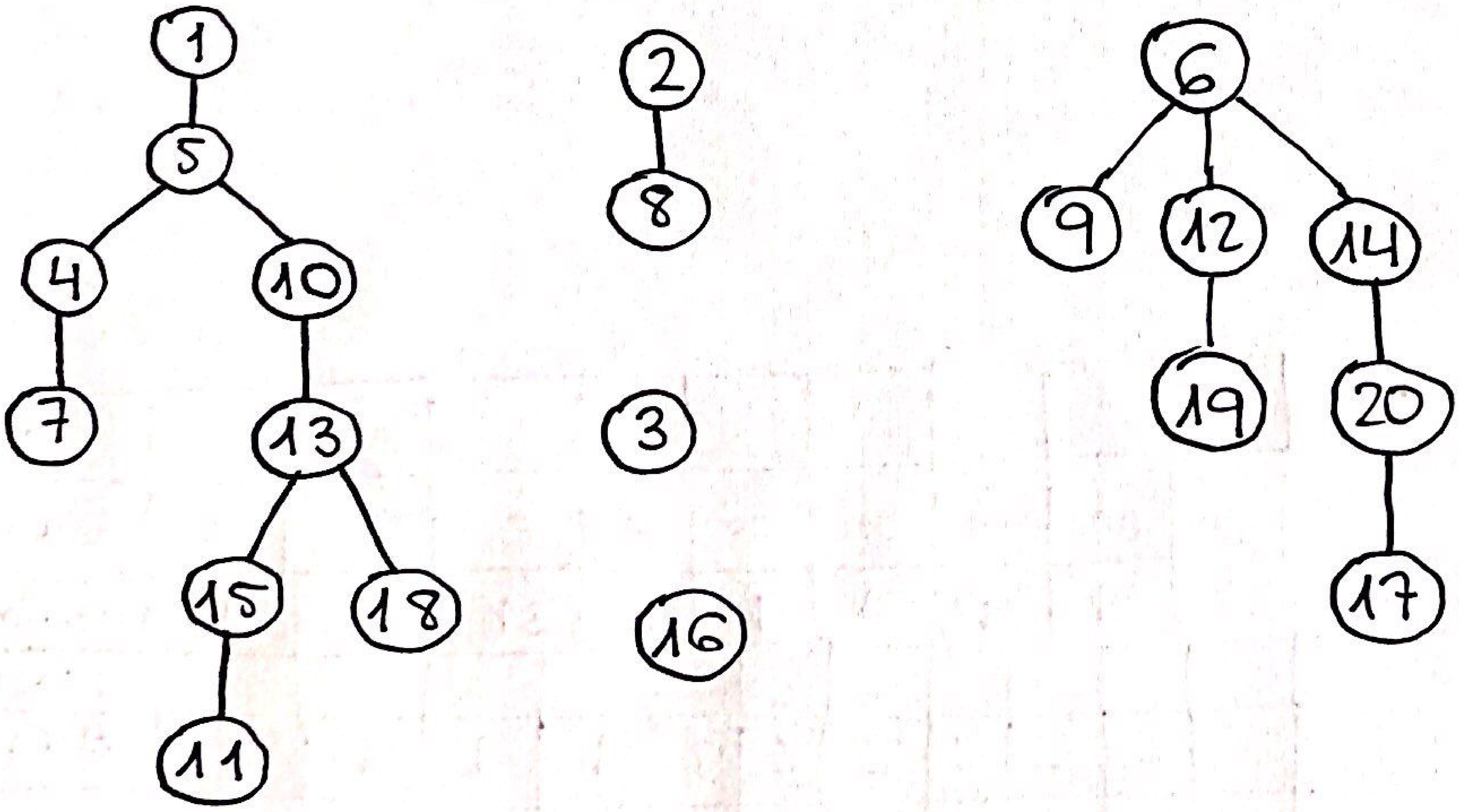
ii)



b) ii)

Topologische Sortierung:

16	6	14	20	17	12	19	9	3	2	8	1	5	10	18	13	15	11	4	7



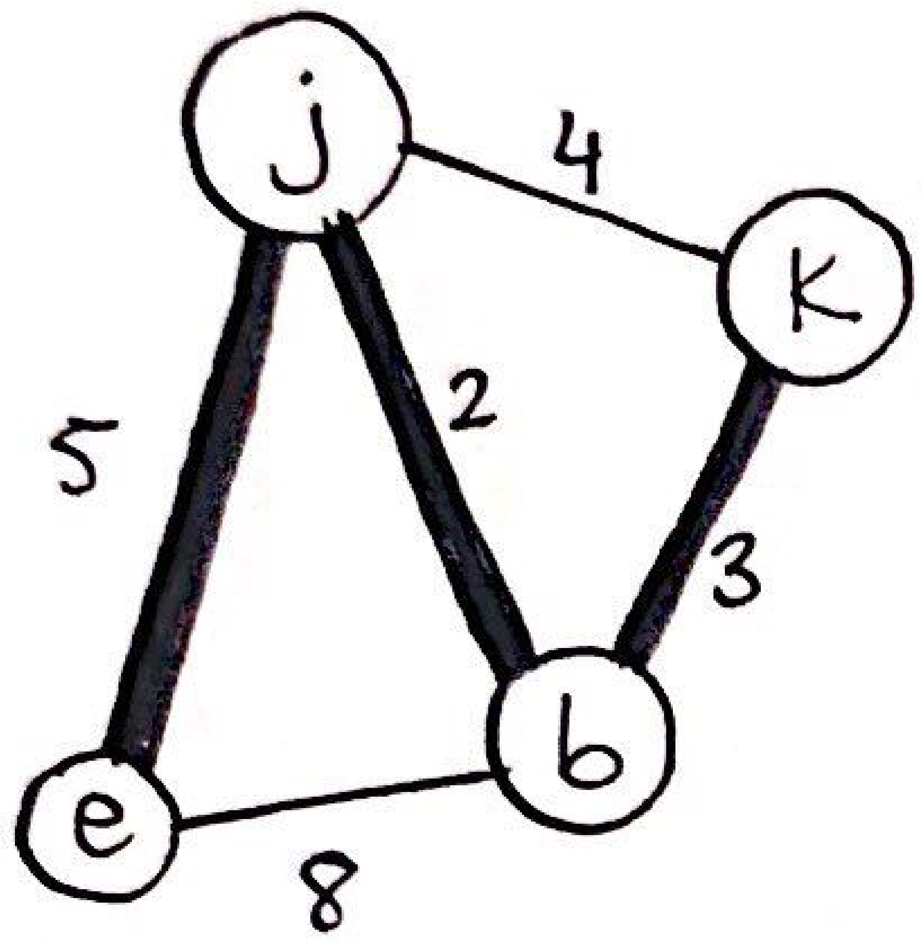
b) i)

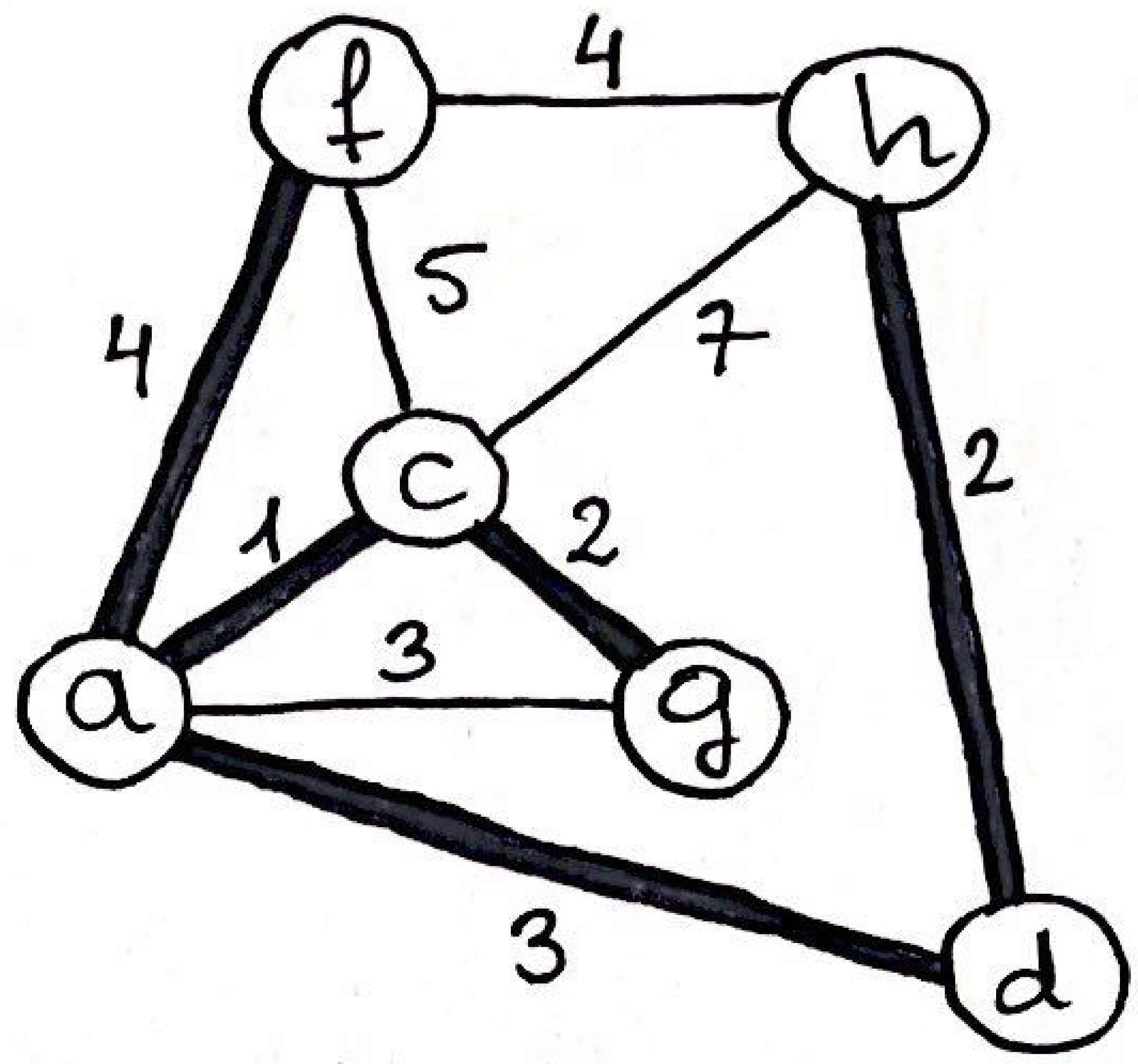
Knoten	Entdeckungszeit	Abschlusszeit	Vorgängerknoten
1	1	18	nil
2	19	22	nil
3	23	24	nil
4	3	6	5
5	2	17	1
6	25	38	nil
7	4	5	4
8	20	21	2
9	26	27	6
10	7	16	5
11	10	11	15
12	28	31	6
13	8	13	10
14	32	37	6
15	9	12	13
16	39	40	nil
17	34	35	20
18	14	15	13
19	29	30	12
20	33	36	14

H2: KRUSKAL UND PRIM

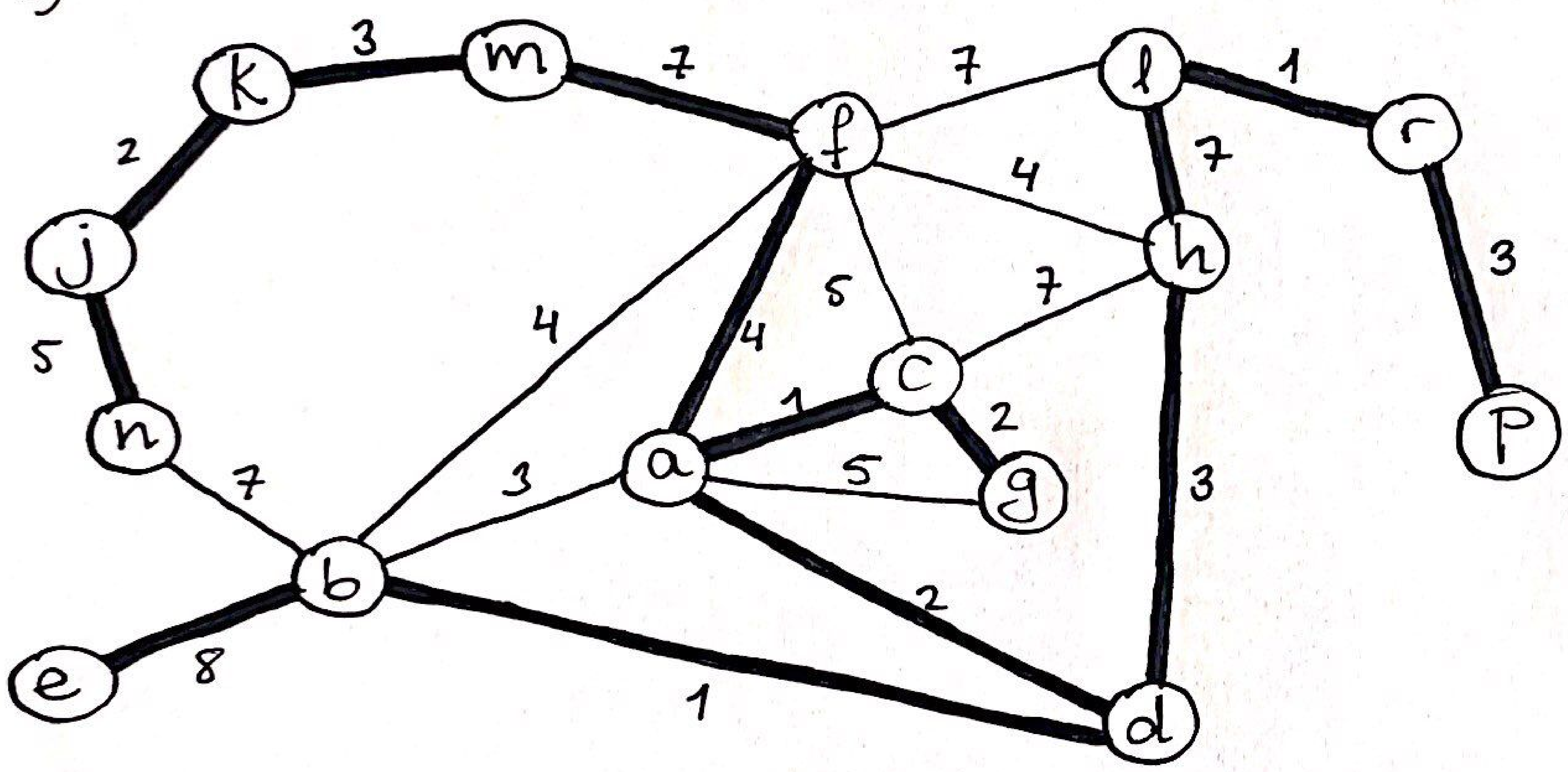
$\{u,v\}$	$w(\{u,v\})$	Dazu?	set(a)	set(b)	set(c)	set(d)	set(e)	set(f)	set(g)	set(h)	set(j)	set(k)
□	□	□	$\{a\}$	$\{b\}$	$\{c\}$	$\{d\}$	$\{e\}$	$\{f\}$	$\{g\}$	$\{h\}$	$\{j\}$	$\{k\}$
{a,c}	1	Ja	{a,c}	=	{a,c}	=	=	=	=	=	=	=
{b,j}	2	Ja	=	{b,j}	=	=	=	=	=	=	{b,j}	=
{c,g}	2	Ja	{a,c,g}	=	{a,c,g}	=	=	=	{a,c,g}	=	=	=
{d,h}	2	Ja	=	=	=	{d,h}	=	=	=	{d,h}	=	=
{a,d}	3	Ja	{a,c,d,g,h}	=	{a,c,d,g,h}	{a,c,d,g,h}	=	=	{a,c,d,g,h}	{a,c,d,g,h}	=	=
{a,g}	3	Nein	=	=	=	=	=	=	=	=	=	=
{b,k}	3	Ja	=	{b,j,k}	=	=	=	=	=	=	{b,j,k}	{b,j,k}
{a,f}	4	Ja	{a,c,d,f,g,h}	=	{a,c,d,f,g,h}	{a,c,d,f,g,h}	=	{a,c,d,f,g,h}	{a,c,d,f,g,h}	{a,c,d,f,g,h}	=	=
{f,h}	4	Nein	=	=	=	=	=	=	=	=	=	=
{j,k}	4	Nein	=	=	=	=	=	=	=	=	=	=
{c,f}	5	Nein	=	=	=	=	=	=	=	=	=	=
{e,j}	5	Ja	=	{b,e,j,k}	=	=	{b,e,j,k}	=	=	=	{b,e,j,k}	{b,e,j,k}
{c,h}	7	Nein	=	=	=	=	=	=	=	=	=	=
{b,e}	8	Nein	=	=	=	=	=	=	=	=	=	=

MST:


j
K
e
b
4
5
2
3
8


f
h
c
a
g
d
4
5
7
4
2
1
2
3
3

b)


K
m
f
l
r
j
h
n
c
a
g
b
e
d
p
3
7
7
1
2
4
7
3
5
4
5
7
4
1
2
7
3
5
3
2
8
1

$a.k$	$b.k$	$c.k$	$d.k$	$e.k$	$f.k$	$g.k$	$h.k$	$j.k$	$k.k$	$l.k$	$m.k$	$n.k$	$p.k$	$r.k$	u
$-\infty$	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	–
=	3	1	2	=	4	5	=	=	=	=	=	=	=	=	a
=	=	=	=	=	=	2	7	=	=	=	=	=	=	=	c
=	1	=	=	=	=	=	3	=	=	=	=	=	=	=	d
=	=	=	=	8	=	=	=	=	=	=	=	7	=	=	b
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	g
=	=	=	=	=	=	=	=	=	=	7	=	=	=	=	h
=	=	=	=	=	=	=	=	=	=	=	7	=	=	=	f
=	=	=	=	=	=	=	=	=	=	=	=	=	=	1	l
=	=	=	=	=	=	=	=	=	=	=	=	=	3	=	r
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	p
=	=	=	=	=	=	=	=	=	3	=	=	=	=	=	m
=	=	=	=	=	=	=	=	2	=	=	=	=	=	=	k
=	=	=	=	=	=	=	=	=	=	=	=	5	=	=	j
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	n
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	e

$a.p$	$b.p$	$c.p$	$d.p$	$e.p$	$f.p$	$g.p$	$h.p$	$j.p$	$k.p$	$l.p$	$m.p$	$n.p$	$p.p$	$r.p$	Q
nil	nil	nil	nil	nil	nil	nil	nil	nil	nil	nil	nil	nil	nil	nil	$\{a,b,c,d,e,f,g,h,j,k,l,m,n,p,r\}$
=	a	a	a	=	a	a	=	=	=	=	=	=	=	=	}b,c,d,e,f,g,h,j,k,l,m,n,p,r{
=	=	=	=	=	=	c	c	=	=	=	=	=	=	=	}b,d,e,f,g,h,j,k,l,m,n,p,r{
=	d	=	=	=	=	=	d	=	=	=	=	=	=	=	}b,e,f,g,h,j,k,l,m,n,p,r{
=	=	=	=	b	=	=	=	=	=	=	=	b	=	=	}e,f,g,h,j,k,l,m,n,p,r{
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	}e,f,h,j,k,l,m,n,p,r{
=	=	=	=	=	=	=	=	=	=	h	=	=	=	=	}e,f,j,k,l,m,n,p,r{
=	=	=	=	=	=	=	=	=	=	=	f	=	=	=	}e,j,k,l,m,n,p,r{
=	=	=	=	=	=	=	=	=	=	=	=	=	=	l	}e,j,k,m,n,p,r{
=	=	=	=	=	=	=	=	=	=	=	=	=	r	=	}e,j,k,m,n,p{
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	}e,j,k,m,n{
=	=	=	=	=	=	=	=	=	m	=	=	=	=	=	}e,j,k,n{
=	=	=	=	=	=	=	=	k	=	=	=	=	=	=	}e,j,n{
=	=	=	=	=	=	=	=	=	=	=	=	j	=	=	}e,n{
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	}e{
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	— (leer)

H3: DREIECKE IN EINEM GRAPHEN

a) Wir nehmen an, dass $a_{ii} = 0 \quad \forall i \in [0....n]$

$$A = \begin{pmatrix} a_{00} & a_{01} & \cdots & a_{0N} \\ a_{10} & a_{11} & \cdots & a_{1N} \\ \vdots & \vdots & & \vdots \\ a_{N0} & a_{N1} & \cdots & a_{NN} \end{pmatrix}$$

Mit der Definition von der Matrix Multiplikation erhalten wir:

$$A^2 = \begin{pmatrix} \sum_{n=0}^{N} a_{0n} a_{n0} & \sum_{n=0}^{N} a_{0n} a_{n1} & \cdots & \sum_{n=0}^{N} a_{0n} a_{nN} \\ \vdots & & & \vdots \\ \sum_{n=0}^{N} a_{Nn} a_{n0} & \cdots & & \sum_{n=0}^{N} a_{Nn} a_{nN} \end{pmatrix}$$

Wenn wir jedes Elemen betrachten, stellen wir fest, dass die Werte alle mögliche Wege mit Distanz 2 Knoten darstellen. ZB für

$$a_{01}^{(2)} = \sum_{n=0}^{N} a_{0n} a_{n1} = \cancelto{0}{a_{00} a_{01}} + \cancelto{0}{a_{01} a_{11}} + a_{02} a_{21} \cdots + a_{0N} a_{N1}$$

$a_{02} a_{21} \rightarrow$ Von 0 bis 1 durch 2

$a_{0N} a_{N1} \rightarrow$ Von 0 bis 1 durch N

Damit die ungleich null sind (also 1), müssen die Wege getrennt gleich eins sein (also die Verbindung muss existieren)

Die Summe gibt uns also die Anzahl von möglichen Wegen mit Distanz 2 Knoten, von i bis j (also mit einem Zwischenknoten) wenn wir nochmal mit A multiplizieren:

$$A^3 = \begin{pmatrix} \sum_{n=0}^{N} a_{0n}^{(2)} a_{n0} & \sum_{n=0}^{N} a_{0n}^{(2)} a_{n1} & \dots & \sum_{n=0}^{N} a_{0n}^{(2)} a_{nN} \\ \vdots & \vdots & & \vdots \\ \sum_{n=0}^{N} a_{Nn}^{(2)} a_{n0} & \sum_{n=0}^{N} a_{Nn}^{(2)} a_{n1} & \dots & \sum_{n=0}^{N} a_{Nn}^{(2)} a_{nN} \end{pmatrix}$$

Wenn wir jetzt zB das Element $a_{01}^{(3)}$ betrachten:

$$a_{01}^{(3)} = \sum_{n=0}^{N} a_{0n}^{(2)} a_{n1} \rightarrow$$

$$= \sum_{n=0}^{N} \left(\sum_{m=0}^{N} a_{0m} a_{mn} \right) a_{n1}$$ → wir erhalten also alle Wege von 0 nach 1 durch m und n $\forall m, n \mid m <> n$ and $m, n \neq 0, 1$

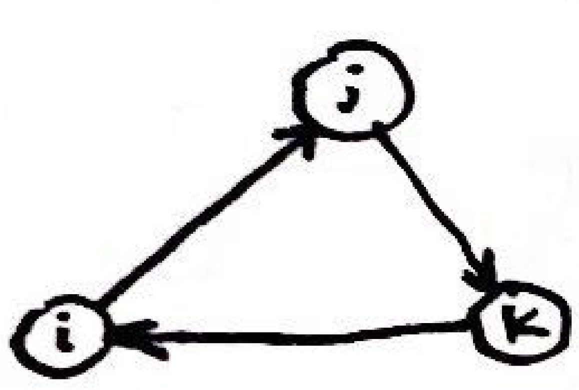
Das sind also alle Anzahl von Wegen von i nach j, die durch andere zwei Knoten gehen.

Für höhere Werte wird es Analog gemacht.

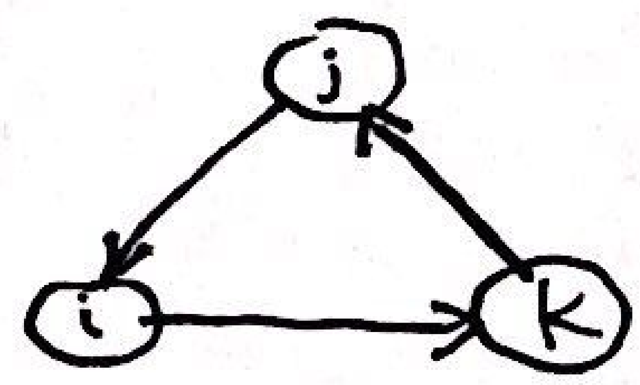
$a_{ij}^{(n)}$ = alle Wege der Länge $(n-1)$, die in i anfangen und irgendwo m enden, mal den Weg von m bis j. Das ist alle Wege der länge n die in i anfangen und in j enden.

b) Mit der matematischen Arbeit schon gemacht, erhalten wir für diesen Algorithmus eine sehr einfache Lösung mittels A^3. Die Elemente $a_{ii}^{(3)}$ $\forall i \in [0 \dots N]$. enthalten die Anzahl von Wegen die in i anfangen und enden, deren Länge 3 Knoten sind. Das sind also alle Mögliche Dreiecke. Wir müssen nur drauf auf passen, die Duplikaten abzuziehen:

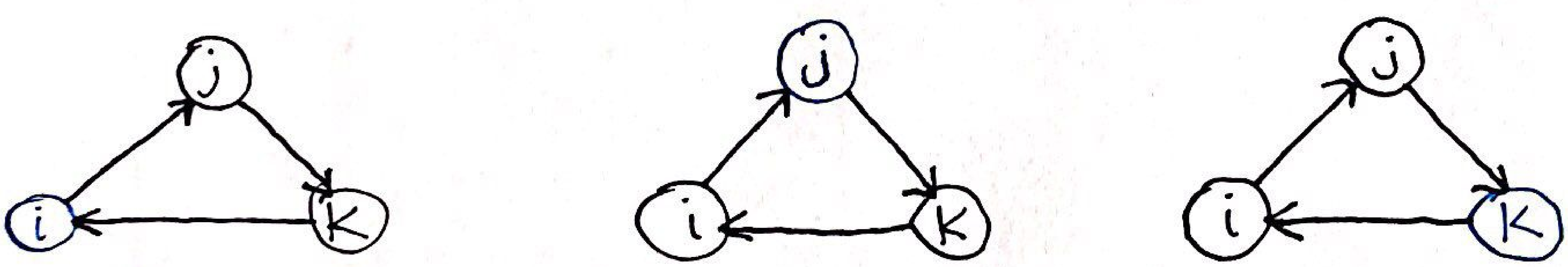
Für ein Anfangsknoten: 1 Duplikat, weil es 2 Richtungen gibt



und



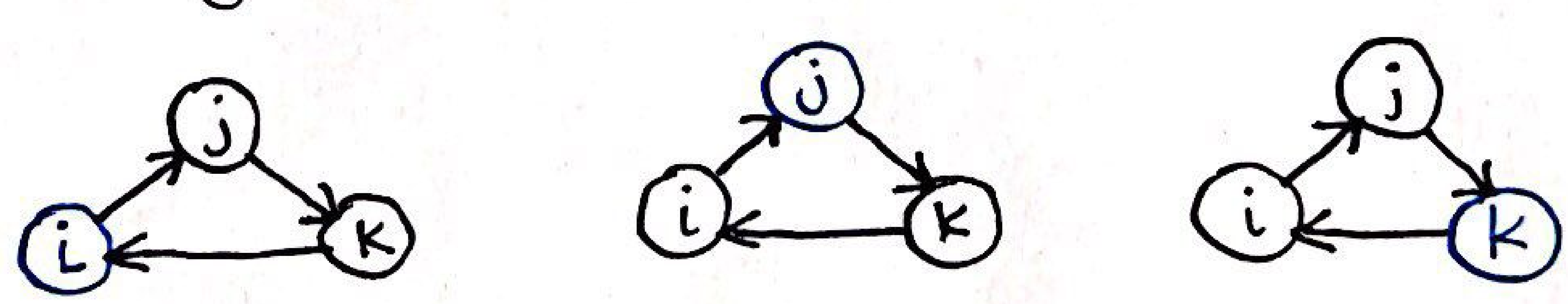
Für jedes Dreieck: 2 Duplikaten, weil es 3 mögliche Anfangsknoten gibt



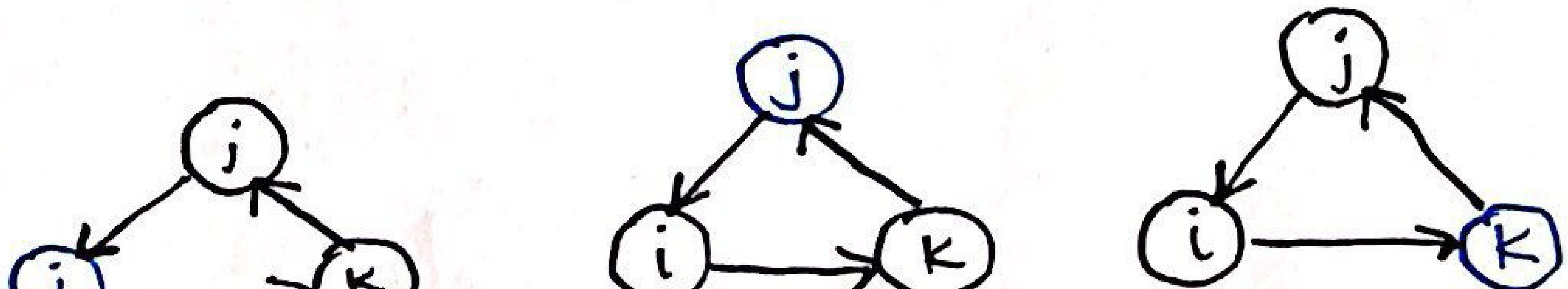
→ Anfangsknoten.

Wir müssen also das Ergebnis durch 6 dividieren (2 * 3)

Für ein Dreieck



6 möglichkeiten.



```
GRAPHTRIANGLES (A)

    A3 = MATRIXPOTENZ (A, 3)
    numDreieck = 0
    FOR i=0 TO A.length - 1
        numDreieck += A3[i][i]
    RETURN numDreieck / 6

MATRIXPOTENZ (A, r)

    let B be a new Array [A.length][A[0].length]
    B = A
    WHILE r > 1
        let C be a new Array [A.length][A[0].length]
        FOR i=0 TO A.length - 1
            FOR j=0 TO A[0].length - 1
                FOR K=0 TO B[0].length - 1
                    C[i][j] += A[i][K] * B[K][j]
        B = C
        r--
    RETURN B
```