

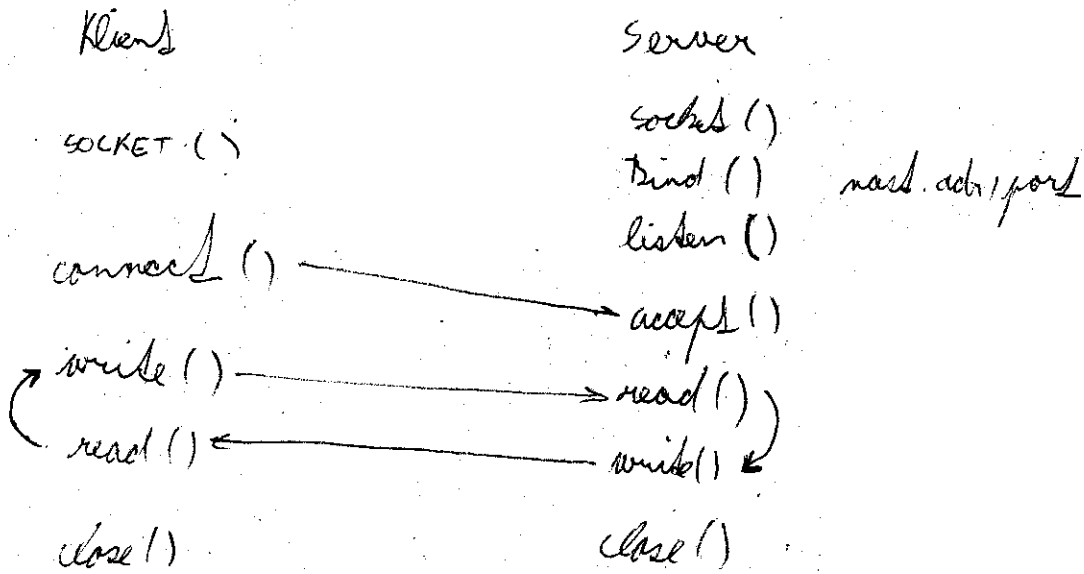
Michal Kostal

kostam1@fel.wz.cz

<http://pitbul.wz.cz>

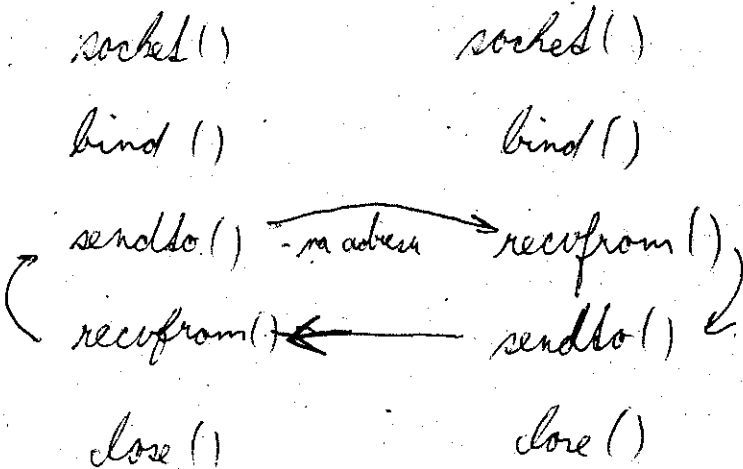
adm.felk.wz.cz/education.cz/X36PK0

TCP



UDP

bez spojení



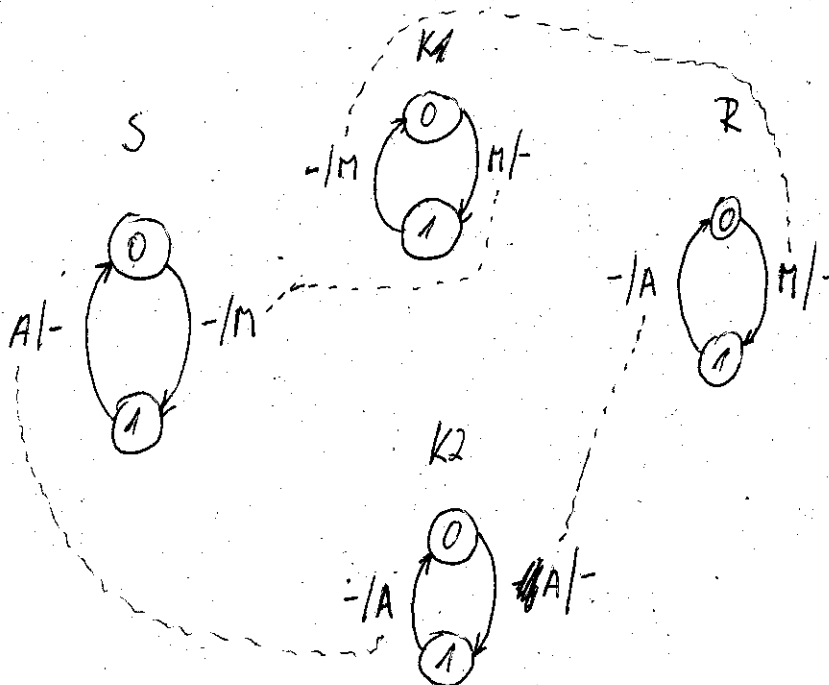
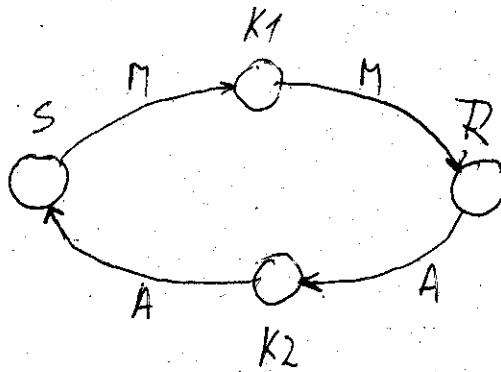
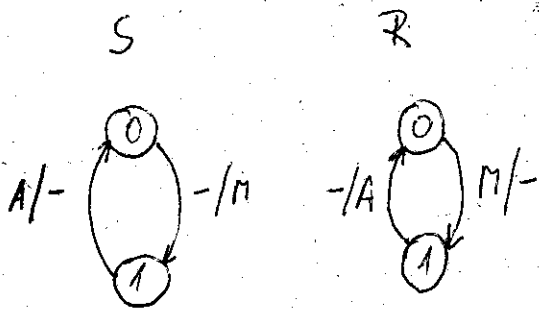
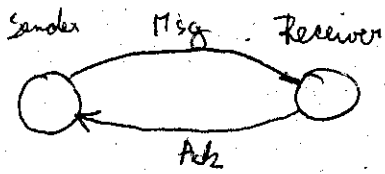
gcc -lsocket -lnsl sep.c

zibkom1@fel.wut.cz

293 561 663

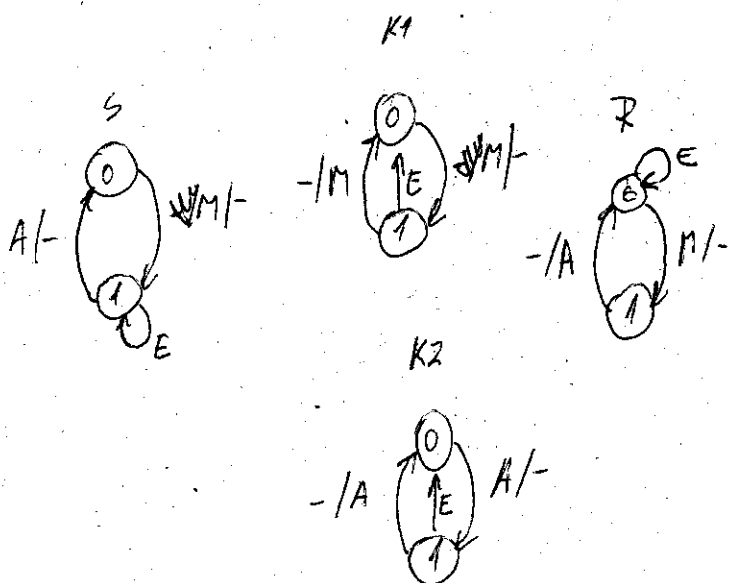
2

Popis protokolu automaty



tento automat nemí mít kone. stav, vždy se lze dostat do počátečního
graf dosažitelnosti

S	K1	R	K2
0	0	0	0
1M	0	0	0
1	1	0	0
1	0M	0	0
1	0	1	0
1	0	0A	0
1	0	0	1
1	0	0	0A
0	0	0	0

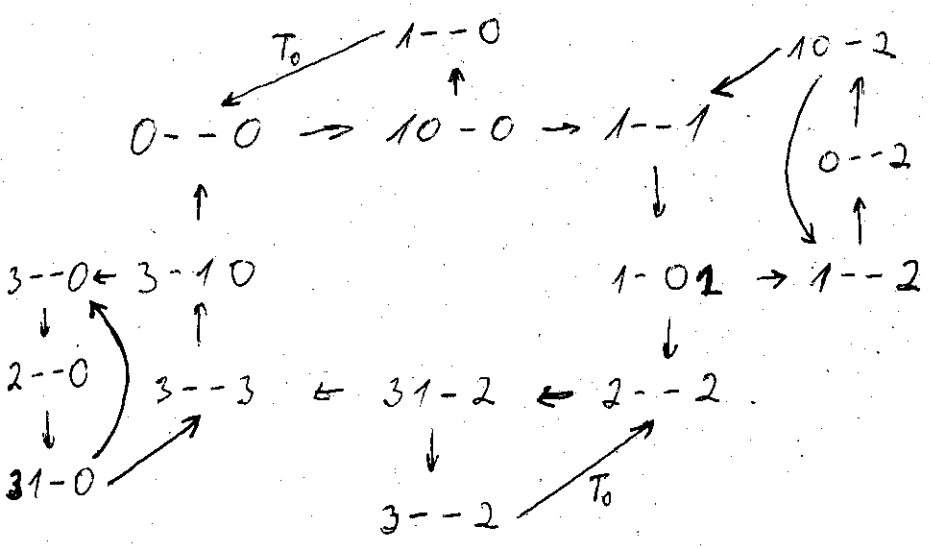
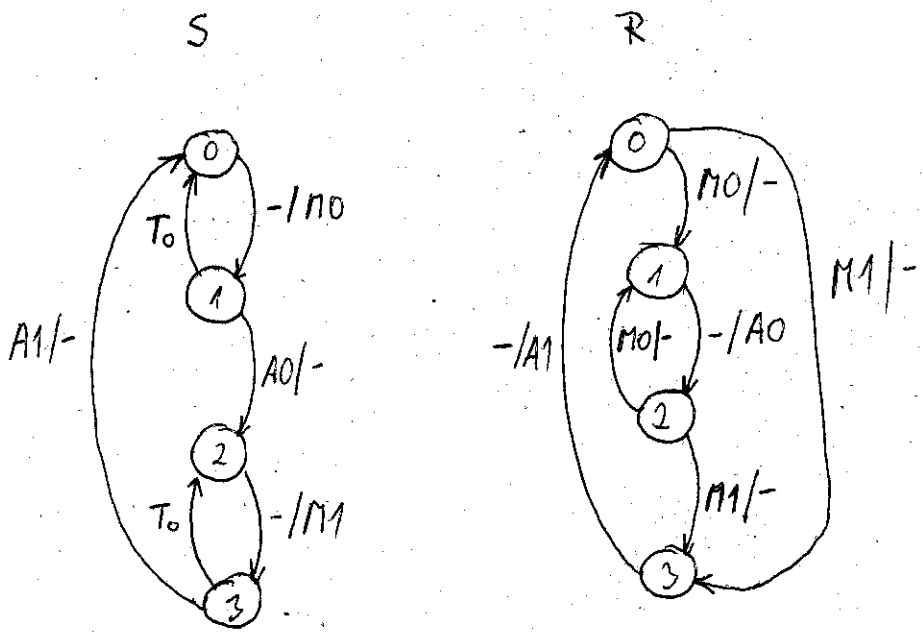


S	K1	R	K2
0	0	0	0
1M	0	0	0
1	1	0	0
1	on	0	0
1	0	1	0
1	0	0A	0
1	0	0	1
1	0	0	0A
0	0	0	0

1 OE 0 0
1 0 0 0

1 0 0 OE
1 0 0 0

radukne



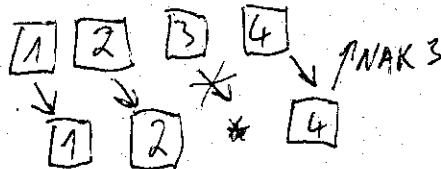
Efektivní protokoly

- řešení špatných paketů

zpětná vazba

- potvrzování - pozitivní ACK
- negativní NAK

+ lineovody



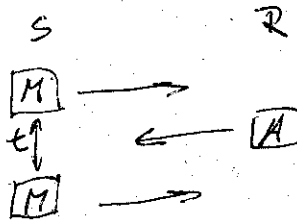
- CRC kód - detekce chyb

- duplikace, změněné pořadí paketů - sekvenční číslovaní

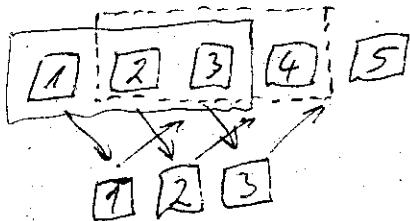
posílovací schémata

• STOP-AND-WAIT

nízká efektivnost
využití linky

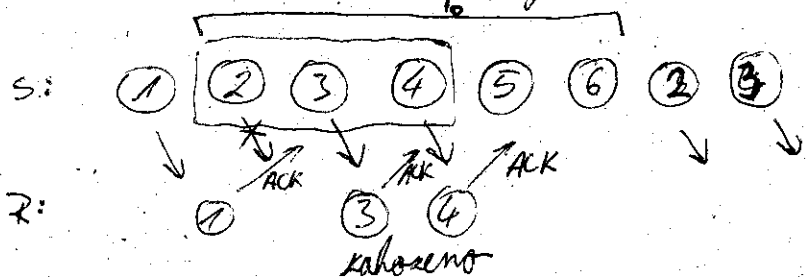


• SLIDING WINDOW

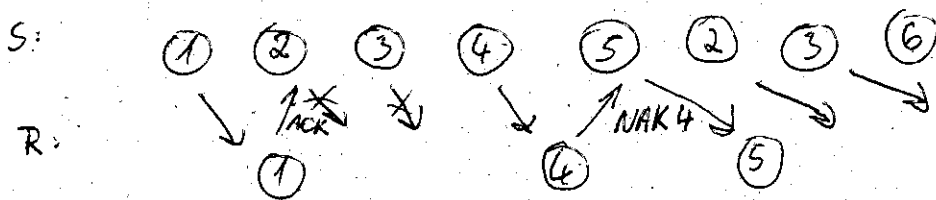


• a) GO-BACK-N

ořadí čeká na další paket,
další nahazuje

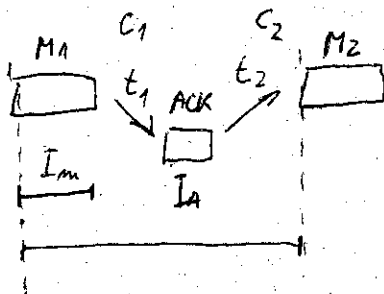


• b) SELECTIVE REPEAT



Receiver má buffer pro pakety mimo pořadí

Efektivita - poměr užitečně využitého pásma
k využitému



c_1, c_2 ... rychlosti linky

$$ef = \frac{L_m}{L_m + t_1 c_1 + L_A + t_2 c_2} = \frac{L_m}{L_m + L_A + 2tc} = \frac{L_m}{L_D}$$

zjednodušeně

Pr $L_m = 1B$
 $c = 10 \text{ Mbit/s}$
 $L = 1 \text{ ms}$
 $L_A = 1B$

$$ef = \frac{8}{8 + 8 + 2 \cdot 10^3 \cdot 10^{-3}} = 0,04 \%$$

$L_m = 1000 B$

$$ef = \frac{1000}{1000 + 8 + 2 \cdot 10^3 \cdot 10^{-3}} = 28,6 \%$$

P - pravděpodobnost poruchy 1 bitu

$$P_N = 1 - (1 - P)^N \quad \text{---||---} \quad \text{---||---} \quad N \text{ bitů}$$

$$ef_P = \frac{l_M}{l_D (1 + P_N (1 + P_N (\dots)))} =$$

... při opakování kvůli chybě může
opět nastat chyba

$$= \frac{l_M}{l_D (1 + P_N + P_N^2 \dots)} = \frac{l_M}{l_D \frac{1}{1 - P_N}} =$$

$$= \frac{l_M}{l_D} \cdot \frac{1}{\frac{1}{1 - P_N}} = \frac{l_M}{l_D} \cdot (1 - P_N)$$

P_N . $l_M = 1000 \text{ B}$

$P = 1 \cdot 10^{-4}$

$$P_N = 1 - (1 - 1 \cdot 10^{-4})^{10000} = 0,55$$

$$ef_P = \frac{28,6}{1} \cdot (1 - 0,55) = 12,8 \%$$

Eff. sliding window

$$ef = M \cdot \frac{l_m}{l_m + l_1 c_1 + l_A + t_2 c_2} = M \cdot \frac{l_m}{l_m + l_A + 2l c} = M \cdot \frac{l_m}{l_0}$$

M - velikost okna

$l_m = 1B$, $t = 1ms$, $c = 10 Mbps$, $l_A = 1B$, $M = 5$

$$ef = 0,2\%$$

$$ef_P = ef_{P=0} (1 - P_N)$$

(6)

příst. konfigurace směrování, projít příslušné materiály Linux i Cisco

server s více klienty

- fork threads.

- select / poll

- non-blocking

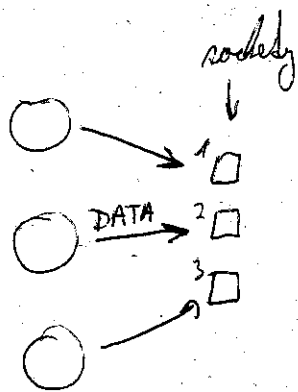
při vytvoření nového spojení nový proces, který ho obsluhuje

ACCEPT
FORK
READ, WRITE

jednodušší pomocí select

```
SELECT (int n, fd_set *read-set,  
        -"- write-set,  
        -"- exception-set)
```

vrátí sockets z množiny, ze kterých se dá číst, příp. zapisovat

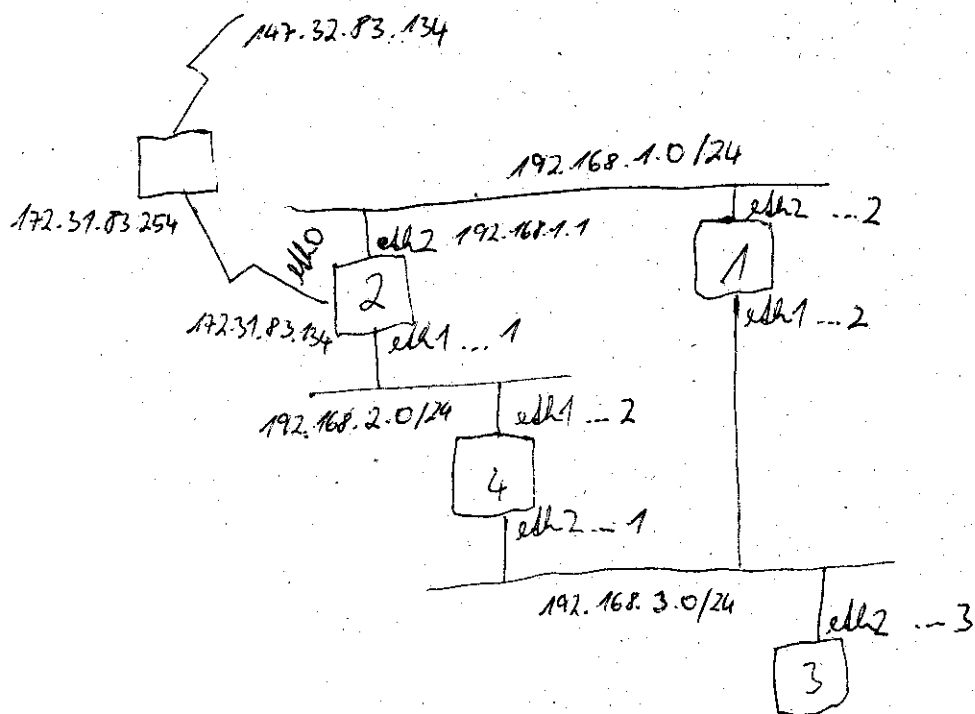


SELECT vrátí socket 2
READ, WRITE může číst bez blokování

socket lze také nastavit pro neblokující volání read, které při nepřítomnosti dat vrátí chybu
použije pouze ve specifických případech

8

přítel Linux od 10³⁰



IP / Maska

192.168.0.0 / 255.255.255.0

~ 192.168.0.0 - 192.168.255.255

192.168.1.0 / 255.255.255.252 nebo 192.168.10/30
~ 192.168.1.0 - 192.168.1.3

1
počet jednotek
v masce

Privátní adresy

10.0.0.0 - 10.255.255.255 10.0.0.0/8
172.16.0.0 - 172.31.255.255 172.16.0.0/12
192.168.0.0 - 192.168.255.255

[2] ifconfig eth2 192.168.1.1 netmask 255.255.255.0
směrování

Dest	G	dev
192.168.3.0/24	192.168.2.2	eth1
192.168.2.0/24	0.0.0.0	eth1

← automaticky

route -en výpis

route add -net 192.168.3.0 netmask 255.255.255.0 gw 192.168.2.2
eth1

-host ... konkrétní adresa
route add default gw ... dev ... co jinak nemá

- 2: route add -net 192.168.3.0 netmask 255.255.255.0 gw 192.168.2.2 eth1
- 3: route add default gw 192.168.3.2 eth2
- 1: - " - net 192.168.2.0 netmask ... gw 192.168.3.1 eth1
- 4: - " default gw 192.168.2.1 eth1
- 1: - " default gw 192.168.1.1 eth2

NAT Network Address Translation

SNAT překlad zdrojové adr.

DNAT

cílové -- před routováním

insert

-I

iptables -t nat

-A
append

PREROUTING

POSTROUTING

-s

zdroj

-d

cíl

-i

nat

-o

výst. interface

-j

typ
překl.

SNAT

DNAT

--to 1.2.3.4

2:

iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

iptables -t nat -A PREROUTING -i eth0 -j DNAT
-d 172.31.83.134 -j DNAT --to 192.168.2.2

→ [2] přeloží adresu z eth0 na adresu [4]

lze překládat i podle portu

pokud je více veřejných adres, lze využít dynamický překlad,
který veřejné adresy přiděluje podle potřeby

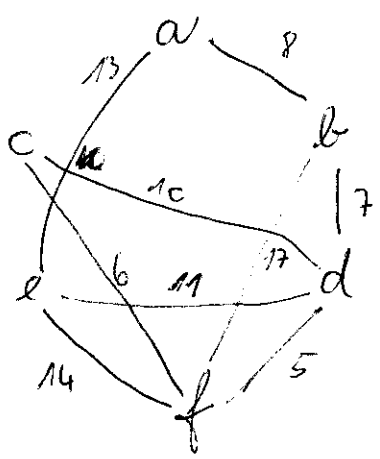
ka 2 dílny písemka

Číslování - dynamické

používají se grafové algoritmy

Dijkstra

u



	V(u)	P	D(u)
a			
b			

delší min early
- přecházíte

ne nelze
chápat

	L(u)	P	D(u)		L(u)	P	D(u)		L(u)	P	D(u)		L(u)	P	D(u)
a	0	1	a	0	1	a	0	1	a	0	1	a	0	1	a
b	∞	0	b	8	1	a	8	1	a	8	1	a	8	1	a
c	∞	0	c	∞	0	c	∞	0	c	∞	0	c	25	0	d
d	∞	0	d	∞	0	d	15	0	b	15	1	b	15	1	b
e	∞	0	e	13	0	a	13	1	a	13	1	a	13	1	a
f	∞	0	f	∞	0	f	25	0	b	25	0	b	20	1	d

Ford-Fulkerson

	L	D		L	D		L	D
a	0	a	0	a	0	a	0	a
b	∞	b	8	a	8	a	8	a
c	∞	c	∞	c	∞	c	25	d
d	∞	d	∞	d	15	b	15	b
e	∞	e	13	a	13	a	13	a
f	∞	f	∞	f	25	b	20	d

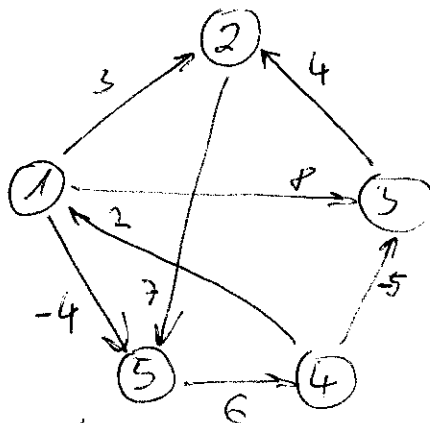
použití v RIP protokolů
uзел si vytvoří sm tabulku
podle tabulek sousedů

v každém kroku vybrána nejkratší cesta k každému uzlu
vzhledem k předchozímu stavu, opakováno dokud se není

protokol OSPF

uzel získa topologii siete a sám si vytvorí
směrovací tabulku

Floyd



$k=0$ nad. sousednosti

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

$$\text{Dist}(k, i, j) = \min \left[\text{Dist}(k-1, i, j), \text{Dist}(k-1, i, k) + \text{Dist}(k-1, k, j) \right]$$

pro nalezení cesty
sítě matice předchozí

$k=1$

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

	1	2	3	4	5
1	N	1	1	N	1
2	N	N	N	N	2
3	N	3	N	N	N
4	4	1	4	N	N
5	N	N	N	5	N

	1	2	3	4	5
1	N	1	1	N	4
2	N	N	N	N	2
3	N	3	N	N	N
4	4	1	4	N	1
5	N	N	N	5	N

$$k=2$$

	1	2	3	4	5
1	0	3	8	∞	-4
2	∞	0	∞	∞	7
3	∞	4	0	∞	11
4	2	5	-5	0	-2
5	∞	∞	∞	6	0

	1	2	3	4	5
1	N	1	1	N	4
2	N	N	N	N	2
3	N	3	N	N	2
4	4	4	N	N	1
5	N	N	N	5	N

$$k=3$$

	1	2	3	4	5
1	0	3	8	∞	-4
2	∞	0	∞	∞	7
3	∞	4	0	∞	11
4	2	-1	-5	0	-2
5	∞	∞	∞	6	0

	1	2	3	4	5
1	N	1	1	N	4
2	N	N	N	N	2
3	N	3	N	N	2
4	4	4	N	N	1
5	N	N	N	5	N

$$k=4$$

	1	2	3	4	5
1	0	3	8	∞	-4
2	∞	0	∞	∞	7
3	∞	4	0	∞	11
4	2	-1	-5	0	-2
5	8	5	1	6	0

	1	2	3	4	5
1	N	1	1	N	4
2	N	N	N	N	2
3	N	3	N	N	2
4	4	4	N	N	1
5	5	4	4	4	5

$$k=5$$

	1	2	3	4	5
1	0	1	-3	2	-4
2	15	0	8	13	7
3	17	4	0	17	11
4	2	-1	-5	0	-2
5	8	5	1	6	0

	1	2	3	4	5
1	N	5	5	5	4
2	5	N	5	5	2
3	17	3	N	5	2
4	4	3	4	N	3
5	4	4	4	5	N

$$5 \rightarrow 2$$

$$5, 4, 3, 2$$