



بسم تعالی

دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

درس شبکه های کامپیوتری، نیمسال دوم سال تحصیلی ۹۶-۹۷

تمرین سری یازدهم (موعده تحویل: ۱۳۹۷/۳/۳۰)



نمره	مسئله
	۹

نمره	مسئله
	۱
	۲
	۳
	۴
	۵
	۶
	۷
	۸

توجه: پاسخ تمرین ها باید به صورت دستنویس تحویل داده شود.

نام و نام خانوادگی:

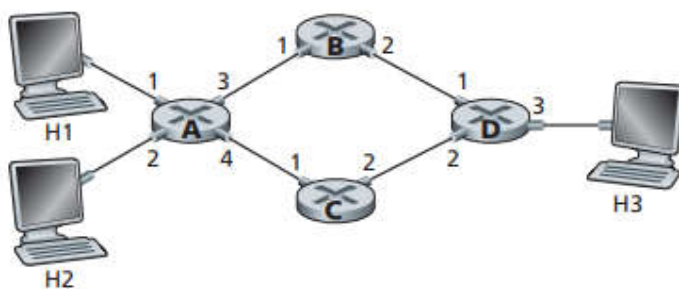
شماره دانشجویی:

نمره:

توجه: برای صرفه جویی در کاغذ تکالیف را یا دو رو پرینت بگیرید و یا از کاغذهای باطله یک رو سفید استفاده کنید.

۱. شبکه زیر را در نظر بگیرید:

- a. فرض کنید این شبکه یک شبکه دیتاگرام است. با این فرض که تمامی ترافیک های مربوط به H3 از طریق interface شماره 3 مسیریاب A ارسال می شوند جدول جلورانی آن را رسم کنید. آیا می توانید جدول جلورانی مربوط به مسیریاب A را طوری بنویسید که تمام ترافیک ارسالی از H1 به H3 از طریق رابط شماره 3 ارسال شود و ترافیک ارسالی از H2 به H3 از طریق رابط شماره 4 ارسال شود؟
- b. اکنون فرض کنید که این شبکه یک شبکه ی مدار مجازی است و یک تماس دائمی بین H1 و H3 وجود دارد و یک تماس دائمی دیگر بین H2 و H3 وجود دارد. جدول جلورانی مربوط به مسیریاب A را بنویسید، به طوری که تمامی ترافیک های ارسال شده از H1 به H3 از طریق رابط شماره 3 و ترافیک ارسالی از H2 به H3 از طریق رابط شماره 4 ارسال شود.
- c. با فرض سناریوی مشابه بخش قبل، جداول جلورانی مربوط به گره های A، B، C و D را بنویسید.



۱. a. داده های ارسالی به مقصد H3 از طریق رابط شماره 3 ارسال می شوند.

آدرس مقصد	رابط
H3	3

با توجه به این که جدول جلورانی در هر مسیریاب با توجه به آدرس مقصد است، پس نمی توان ترافیک ارسالی از H2 به H3 را از طریق رابط شماره 4 ارسال کرد.



(b.۱) توجه داشته باشید که شماره ی VC مربوط به هر دو جریان می تواند یکی باشد.

Incoming interface	Incoming VC#	Outgoing Interface	Outgoing VC#
1	12	3	22
2	63	4	18

(d.۱)

Router B.			
Incoming interface	Incoming VC#	Outgoing Interface	Outgoing VC#
1	22	2	24

Router C.			
Incoming interface	Incoming VC#	Outgoing Interface	Outgoing VC#
1	18	2	50

Router D.			
Incoming interface	Incoming VC#	Outgoing Interface	Outgoing VC#
1	24	3	70
2	50	3	76



۲. فرض کنید دو بسته به صورت همزمان به دو پورت ورودی متفاوت از یک مسیریاب می‌رسند. همچنین فرض کنید که هیچ بسته‌ی دیگری در مسیریاب وجود ندارد.

- فرض کنید که این دو بسته به دو پورت خروجی متفاوت ارسال می‌شوند. آیا می‌توان دو بسته را به طور همزمان از طریق switch fabric ای که از shared-bus استفاده می‌کند، ارسال کرد؟
- فرض کنید که این دو بسته به دو پورت خروجی متفاوت ارسال می‌شوند. آیا می‌توان دو بسته را به طور همزمان از طریق switch fabric ای که از crossbar استفاده می‌کند، ارسال کرد؟
- فرض کنید که این دو بسته به یک پورت خروجی یکسان ارسال می‌شوند. آیا می‌توان دو بسته را به طور همزمان از طریق switch fabric ای که از crossbar استفاده می‌کند، ارسال کرد؟

- خیر، در هر لحظه فقط می‌توان یک بسته را از طریق shared-bus ارسال کرد.
- بله، تا زمانی که دو بسته از bus ورودی و خروجی متفاوتی استفاده کنند، می‌توانند به صورت موازی ارسال شوند.
- خیر، ارسال دو بسته از طریق bus خروجی یکسان به صورت همزمان امکان پذیر نیست.

۳. یک شبکه دیتاگرام که از آدرس ۳۲ بیتی استفاده می‌کند را در نظر بگیرید. مسیریابی را با چهار لینک در نظر بگیرید که از 0 تا 3 شماره‌گذاری شده است و بسته‌ها به واسطه‌های مربوط به لینک‌ها به صورت زیر ارسال می‌شوند:

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 11111111 11111111	1
11100000 01000001 00000000 00000000 through 11100001 01111111 11111111 11111111	2
otherwise	3

- جدول جلورانی که شامل پنج سطر باشد و از قاعده Longest Prefix Match استفاده می‌کند را طراحی کنید.
- توضیح دهید که چگونه این جدول جلورانی، خروجی مناسب را برای دیتاگرام‌هایی با آدرس مقصدهای زیر انتخاب می‌کند:

```
11001000 10010001 01010001 01010101
11100001 01000000 11000011 00111100
11100001 10000000 00010001 01110111
```



a)

Prefix Match	Link Interface
11100000 00	0
11100000 01000000	1
11100000	2
11100001 1	3
otherwise	3

دقت کنید که اولین آدرس بعد از

11100001 01111111 11111111 11111111

آدرس

11100001 01111111 11111111 11111111

است که در سطر چهارم جدول قرار داده شده است. آدرس هایی که با این سطر Match بشوند باید از واسط سوم خارج شوند. حال با خیال راحت می توان آدرس 1110000 را در سطر چهارم قرار داد: بسته هایی که با آدرس 11100000 00 تطبیق پیدا می کنند از واسط صفرم خارج می شوند. بعد از این آدرس، آدرس 11100000 01 وجود دارد که البته همه این فضا به یک واسط خروجی هدایت نمی شوند. بلکه اگر بسته با 11100000 01000000 تطبیق پیدا کرد از واسط اول خارج می شود. بعد از این آدرس شبکه 11100000 01000001 شروع می شود که دقیقا شروع آدرس هایی است که باید از واسط شماره دوم خارج شود. ما کل بسته هایی که با الگوی 1110000 تطابق پیدا کند را به واسط شماره دوم هدایت میکنیم مگر اینکه با آدرس 11100001 1 تطابق پیدا کند که در آن صورت از واسط سوم خارج می شود.

(b)

پیشوند مطابق پنجمین ردیف جدول است، پس از طریق رابط شماره ی 3 ارسال می شود.

پیشوند مطابق سومین ردیف جدول است، پس از طریق رابط شماره ی 2 ارسال می شود.

پیشوند مطابق چهارمین ردیف جدول است، پس از طریق رابط شماره ی 3 ارسال می شود.



۴. یک مسیریاب که اتصال ۳ زیرشبکه Subnet 1، Subnet 2 و Subnet 3 را برقرار می‌کند، در نظر بگیرید. فرض کنید همه‌ی واسط‌ها در هر یک از این سه زیرشبکه باید پیشوند 223.1.17/24 را داشته باشند. همچنین فرض کنید که زیرشبکه ۱ از ۶۰ میزبان، زیرشبکه ۲ از ۹۰ میزبان و زیرشبکه ۳ از ۱۲ میزبان پشتیبانی می‌کند. سه آدرس شبکه (به فرم a.b.c.d / x محاسبه کنید که محدودیت‌های گفته شده را برآورده کند.

زیرشبکه اول، نیاز به ۶ بیت فضا (۲+۶۰) برای آدرس دهی میزبان‌ها دارد. زیرشبکه دوم به ۷ بیت و زیرشبکه سوم به ۴ بیت فضا نیاز دارد. کل فضای در دسترس هم برای آدرس دهی ۲۵۴ میزبان است.

Subnet 1 230.1.17.128/26 → 223.1.17.100000000/26 → 223.1.17.128 تا 223.1.17.191 = 64

Subnet 2 230.1.17.0/25 → 223.1.17.00000000/25 → 223.1.17.0 تا 223.1.17.127 = 128

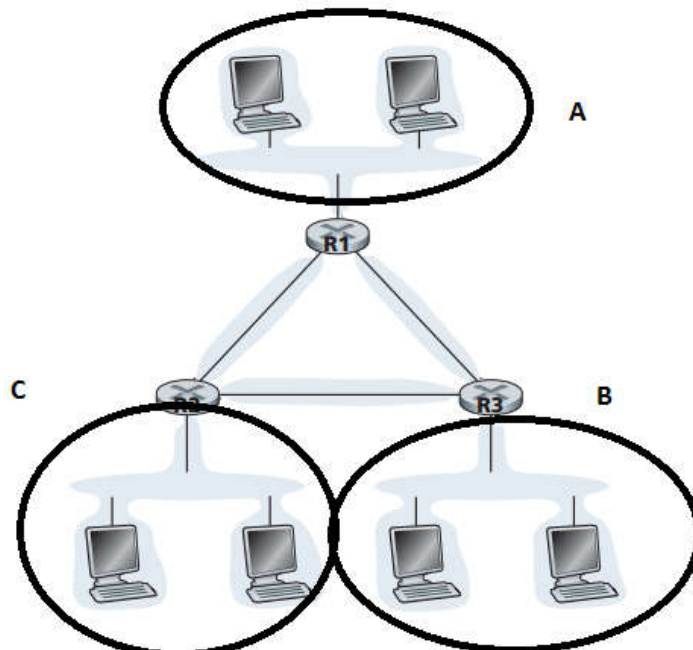
Subnet 3 230.1.17.192/28 → 223.1.17.11000000/28 → 223.1.17.192 تا 223.1.17.207 = 16

آدرس دهی‌های دیگری نیز می‌تواند صحیح باشد به شرطی که آدرس ماشین‌ها در زیرشبکه‌های مختلف با یکدیگر همپوشانی نداشته باشند.

۵. همبندی نشان‌داده شده در شکل زیر را در نظر بگیرید. زیر شبکه‌های دارای میزبان موجود در شکل را به ترتیب ساعتگرد، A، B و C بنامید. زیر شبکه‌های بدون میزبان را نیز به ترتیب D، E و F بنامید.

a. به زیر شبکه‌های A، B، C آدرس شبکه‌ای با این محدودیت‌ها اختصاص دهید: تمام آدرس‌ها باید از فضای 214.97.254/23 اختصاص داده شوند. زیرشبکه A باید آدرس‌های کافی برای 250 واسط داشته باشد. زیرشبکه B باید آدرس‌های کافی برای حمایت از 120 واسط داشته باشد. زیر شبکه C باید آدرس‌های کافی برای حمایت از 120 واسط داشته باشد و زیرشبکه‌های D، E، F هرکدام باید بتوانند دو واسط را پشتیبانی کنند. برای هر واسط آدرس‌های اختصاص داده شده باید به صورت a.b.c.d/x و یا به صورت بازه e.f.g.h/y - a.b.c.d/x باشد.

b. با توجه به پاسخ قسمت a، جدول جلورانی مربوط به سه مسیریاب را با توجه به قانون Longest Prefix Match آماده کنید.



a.

زیرشبکه‌ی A: 214.97.255/24

زیرشبکه‌ی C: 214.97.254.128/25

زیرشبکه‌ی D: 214.97.254.0/31

زیرشبکه‌ی E: 214.97.254.2/31

زیرشبکه‌ی F: 214.97.254.4/30

ممکن است برای زیرشبکه‌های D، E و F شما ۲+۲ کرده باشید و بخواهید Prefix مربوطه را 30/ تعریف کنید. البته این روال کاملاً درست است ولی باعث هدر رفت فضای IP می‌شود و ممکن است نتوانید آدرس دهی را انجام دهید. بنابراین در RFC3021 تصریح شده که برای شبکه‌هایی مانند D و E و F که از یک لینک نقطه به نقطه تشکیل شده است نیازی به اضافه کردن ۲ نداریم و می‌توان از آدرس 31/ برای شماره گذاری دو مسیر یاب استفاده کرد. چگونگی مدیریت کردن Broadcast در RFC بیان شده که خارج از محدوده درس است. با همین فرض می‌توان در نظر گفت که در زیرشبکه B هم همه آدرس‌ها استفاده می‌شود. در این صورت زیرشبکه B از آدرس 214.97.254.8 تا 214.97.254.127 را در اختیار دارد که یک فضا با ۱۲۰ آدرس است. این فضا را می‌توان با چهار آدرس زیر مشخص کرد:

214.97.254.8/29

214.97.254.16/28

214.97.254.32/27

214.97.254.64/26

که تجمیع شده آن 214.97.254.0/25 است. با توجه به این فرضیات:



b. برای مسیریاب اول داریم:

Prefix Match	مقصد	Outgoing Interface
11010110 01100001 11111111	زیر شبکه A	Subnet A
11010110 01100001 11111110 0000000	زیر شبکه D	Subnet D
11010110 01100001 11111110 000001	زیر شبکه F	Subnet F
11010110 01100001 11111110 0000001	زیر شبکه E	Subnet D (دلخواه است)
11010110 01100001 11111110 1	زیر شبکه C	Subnet F
11010110 01100001 11111110 0	کل زیر شبکه های D, B, E و F که طبق LPM اینجا صرفا با B مطابق دارد.	Subnet D

به جای Outgoing Interface می توان اینترفیس ها را شماره گذاری کرد و شماره آن ها را قرار داد. همچنین به جای سه سطر آخر هم می توان یک Default Route به صورت 0.0.0.0/0 قرار داد و از یکی از واسطه های مرتبط با Subnet D و یا F به دلخواه خارج کرد! تا به گره بعدی برسد و تصمیم گیری در آنجا انجام شود.

برای مسیریاب دوم جدول مسیریابی به صورت زیر است:

Prefix Match	مقصد	Outgoing Interface
11010110 01100001 11111111	زیر شبکه A	Subnet D
11010110 01100001 11111110 0000000	زیر شبکه D	Subnet D
11010110 01100001 11111110 000001	زیر شبکه F	Subnet E
11010110 01100001 11111110 0000001	زیر شبکه E	Subnet E
11010110 01100001 11111110 1	زیر شبکه C	Subnet E
11010110 01100001 11111110 0	کل زیر شبکه های D, B, E و F که طبق LPM اینجا صرفا با B مطابق دارد.	Subnet B

برای مسیریاب سوم جدول مسیریابی به صورت زیر است:



Prefix Match	مقصد	Outgoing Interface
11010110 01100001 11111111	زیر شبکه A	Subnet F
11010110 01100001 11111110 0000000	زیر شبکه D	Subnet E
11010110 01100001 11111110 000001	زیر شبکه F	Subnet F
11010110 01100001 11111110 0000001	زیر شبکه E	Subnet E
11010110 01100001 11111110 1	زیر شبکه C	Subnet C
11010110 01100001 11111110 0	کل زیر شبکه های E, D و F که طبق LPM اینجا صرفا با B مطابق دارد.	Subnet E

همچنین می توانید آدرس ها را به صورت زیر نیز اختصاص دهید:

زیر شبکه‌ی A: 214.97.255/24 (256 addresses)

زیر شبکه‌ی D: 214.97.254.0/31 (2 addresses)

زیر شبکه‌ی E: 214.97.254.2/31 (2 addresses)

زیر شبکه‌ی F: 214.97.254.128/31 (4 addresses)

زیر شبکه B: آدرس های 214.97.254.0 تا 214.97.254.3 برای زیر شبکه های D و E استفاده شده اند.

214.97.254.4/30

214.97.254.8/29

214.97.254.16/28

214.97.254.32/27

214.97.254.64/26

که تجميع شده آن همان 214.97.254.0/25 است.

زیر شبکه C: آدرس 214.97.254.128 و 214.97.254.129 برای زیر شبکه F انتخاب شده است. پس زیر شبکه های زیر

موجود هستند:

214.97.254.130/31

214.97.254.132/30

214.97.254.136/29

214.97.254.144/28

214.97.254.160/27

214.97.254.192/26

که تجميع شده آن همان 214.97.254.128/25 است. آدرس های تجميع شده را در جداول مسیریابی استفاده کنید و

جداول مسیریابی را دوباره ایجاد کنید. مسیریابی طبق قاعده Longest Prefix Match به صورت صحیح انجام خواهد شد.

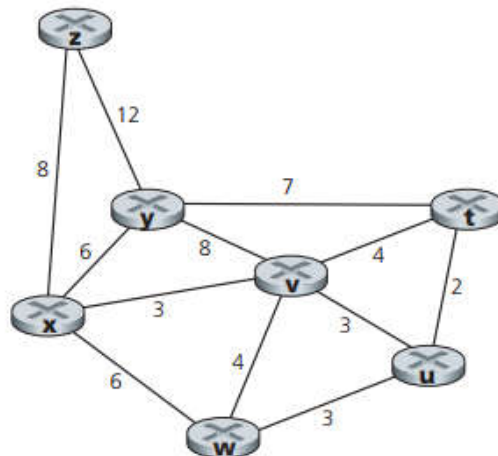


۶. فرض کنید یک مسیر یاب یک بسته‌ی IP که شامل ۶۰۰ بایت داده است را دریافت می‌کند و باید بسته را به یک شبکه با ۲۰۰ MTU بایت ارسال کند. فرض کنید که سرآیند این بسته ۲۰ بایت است. fragment های ایجاد شده توسط مسیر یاب را نشان داده و مقادیر مربوط به سرآیند هر fragment را نشان دهید (برای مثال total length, fragment offset, and more bit).

۲۰ بایت از فضای MTU به سرآیند بسته تعلق دارد. پس ۱۸۰ بایت داده‌ی اصلی در هر بسته می‌توان ارسال کرد. نزدیک‌ترین عدد بخش پذیر به ۸ برابر با ۱۷۶ است. بنابراین بسته به ۴ fragment شکسته می‌شود:

شماره بسته	Total length	Fragment offset	More flag
Original Packet	600+20	0	0
1	176+20	0	1
2	176+20	22	1
3	176+20	44	1
4	72+20	66	0

۷. شبکه‌ی زیر را در نظر بگیرید. با توجه به هزینه‌ی مشخص شده برای لینک‌ها، با استفاده از الگوریتم کوتاه‌ترین مسیر Dijkstra، کوتاه‌ترین مسیر را از X به همه‌ی گره‌های شبکه محاسبه کنید. با استفاده از جدولی مشابه جدول ۴.۳ کتاب نشان دهید که الگوریتم چگونه کار می‌کند.



Step	N'	D(t),p(t)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	x	∞	∞	3,x	6,x	6,x	8,x
1	xv	7,v	6,v		6,x	6,x	8,x
2	xvu	7,v			6,x	6,x	8,x
3	xvuw	7,v				6,x	8,x
4	xvuwy	7,v					8,x
5	xvuwyt						8,x
6	xvuwytz						



۸. همبندی شکل ۴-۳۱ کتاب را در نظر بگیرید. فرض کنید که مسیر یاب دیگری به نام w به مسیر یاب‌های y , z متصل است. هزینه‌ی مربوط به لینک‌ها به این صورت است: $c(x, y) = 4$, $c(x, z) = 50$, $c(y, w) = 1$, $c(z, w) = 1$, $c(y, z) = 3$. فرض کنید که از poisoned reverse در الگوریتم مسیر یابی distance vector استفاده می‌شود.
- a. پس از همگرایی الگوریتم مسیر یابی distance vector، مسیر یاب w ، y و z فاصله‌های خود تا x را به یکدیگر اطلاع می‌دهند. مقدار آن‌ها را مشخص کنید.
- b. حال فرض کنید که هزینه لینک بین x و y به ۶۰ افزایش می‌یابد. آیا در صورت استفاده از poisoned reverse مشکل count-to-infinity وجود دارد؟ چرا؟ اگر این مشکل وجود دارد، مسیر یابی distance vector به چه تعداد تکرار نیاز دارد تا دوباره به حالت پایدار برسد؟ پاسخ خود را توضیح دهید.

a.

Router z	Informs w, $D_z(x) = \infty$
	Informs y, $D_z(x) = 6$
Router w	Informs y, $D_w(x) = \infty$
	Informs z, $D_w(x) = 5$
Router y	Informs w, $D_y(x) = 4$
	Informs z, $D_y(x) = 4$



b. بله، این مشکل وجود دارد. جدول زیر مراحل مربوط به همگرایی مسیریابی را مشخص می‌کند. فرض کنید که در لحظه‌ی t_0 ، هزینه‌ی لینک تغییر می‌کند. در لحظه‌ی t_1 ، y هزینه‌ی لینک‌ها را به‌روزرسانی می‌کند و به همسایه‌های خود (w, z) اطلاع می‌دهد. در جدول زیر '→' نشان دهنده اطلاعاتی است که به گره دیگر رسیده است.

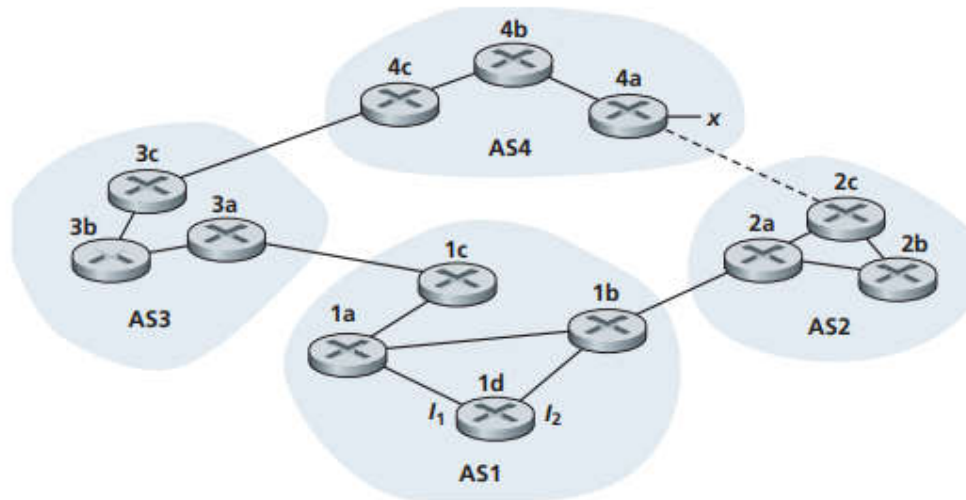
time	t_0	t_1	t_2	t_3	t_4
Z	→ w, $D_z(x)=\infty$ → y, $D_z(x)=6$		No change	→ w, $D_z(x)=\infty$ → y, $D_z(x)=11$	
W	→ y, $D_w(x)=\infty$ → z, $D_w(x)=5$		→ y, $D_w(x)=\infty$ → z, $D_w(x)=10$		No change
Y	→ w, $D_y(x)=4$ → z, $D_y(x)=4$	→ w, $D_y(x)=9$ → z, $D_y(x)=\infty$		No change	→ w, $D_y(x)=14$ → z, $D_y(x)=\infty$

همان‌طور که مشاهده می‌کنید، گره‌های y, w, z برای محاسبه‌ی هزینه تا مسیریاب x یک حلقه را ایجاد می‌کنند. اگر به این تکرار ادامه دهیم، همان‌طور که در جدول بالا مشاهده می‌کنید، در لحظه‌ی t_{27} ، z متوجه می‌شود که کمترین هزینه‌اش به x با یک لینک مستقیم، برابر با 50 است. در لحظه‌ی t_{29} ، کوتاه‌ترین مسیر از w به x از طریق z برابر با 51 است و در لحظه‌ی t_{30} ، کوتاه‌ترین مسیر از y به x از طریق w برابر با 52 است. در نهایت در لحظه‌ی t_{31} هیچ به‌روزرسانی وجود ندارد.

time	t_{27}	t_{28}	t_{29}	t_{30}	t_{31}
Z	→ w, $D_z(x)=50$ → y, $D_z(x)=50$				via w, ∞ via y, 55 via z, 50
W		→ y, $D_w(x)=\infty$ → z, $D_w(x)=50$	→ y, $D_w(x)=51$ → z, $D_w(x)=\infty$		via w, ∞ via y, ∞ via z, 51
Y		→ w, $D_y(x)=53$ → z, $D_y(x)=\infty$		→ w, $D_y(x)=\infty$ → z, $D_y(x)=52$	via w, 52 via y, 60 via z, 53



۹. شبکه زیر را در نظر بگیرید. فرض کنید که AS2 و AS3 برای پروتکل مسیریابی intra-AS از OSPF استفاده می‌کند و AS1 و AS4 برای پروتکل مسیریابی intra-AS از RIP استفاده می‌کند. همچنین فرض کنید که در پروتکل مسیریابی inter-AS از eBGP و iBGP استفاده شده است. از ابتدا فرض شده است که هیچ لینک فیزیکی بین AS1 و AS4 وجود ندارد.



- a) eBGP
- b) iBGP
- c) eBGP
- d) iBGP