



دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

عنوان:

شبکه‌های کامپیوتری - تمرین چهارم

Computer Networks - HW4

شماره درس

۴۰۴۴۳

تاریخ تحویل

۱۴۰۴/۰۳/۳۰

مباحث

فصل پنجم - لایه شبکه - واحد کنترل

استاد درس

دکتر سید امیر مهدی صادق زاده مسگر

نیم سال دوم سال تحصیلی ۱۴۰۳-۱۴۰۴

پیش از پاسخ به سوالات، به نکات زیر توجه فرمایید.

- خروجی تمرین شما می‌بایست یک فایل ZIP باشد.
- پاسخ‌های بخش عملی و نظری می‌بایست طبق استاندارد زیر در فایل PDF آورده شده باشد.
- لازم به ذکر است که اگر در سوالات بخش عملی از شما کدی خواسته شود آن را در دایرکتوری‌های مجزا، طبق استاندارد زیر قرار دهید.

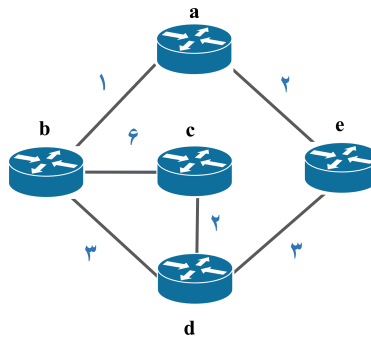
```
CN_HW#_STDID.pdf <--|
(DIR) Practical Section Codes <--|
      (DIR) Q# <--|
      Server.py <--|
      Client.py <--|
      Report.pdf <--|
      Others <--|
```

- اشکالات یا ابهامات خود را از طریق تالار پرسش و پاسخ در نظر گرفته شده برای تمرین مطرح نمایید.
- هر گونه نسخه‌برداری از تمرین‌های دیگران منجر به از دست رفتن نمره تمرین خواهد شد.
- در مجموع ۱۰ روز تاخیر مجاز خواهید داشت که برای هر تمرین ۳ روز را می‌توانید استفاده کنید.
- در صورت استفاده از هر گونه منبع برای پاسخ به سوالات، ذکر اسم و نشانی دقیق و آدرس دسترسی به صفحه مورد نظر الزامی است.
- بارم‌بندی سوالات به قرار زیر است.

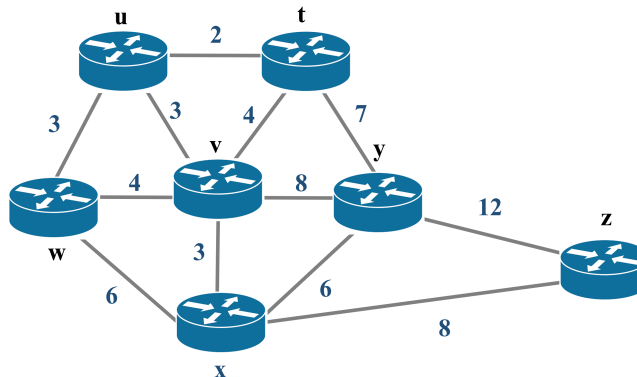
بخش	سوال	بارم
سوالات نظری	سوال ۱	۱۵
	سوال ۲	۱۰
	سوال ۳	۱۵
	سوال ۴	۲۰
سوالات عملی	سوال ۵	۲۰
	سوال ۶	۲۰
جمع نمرات		۱۰۰

سوالات نظری

۱. (۱۵ نمره) شبکه نشان داده شده زیر را در نظر بگیرید. فرض کنید هر گره در ابتدا، هزینه رسیدن به هر یک از همسایگان خود را می‌داند.



- (آ) (۷ نمره) الگوریتم مسیریابی بردار فاصله^۱ را در نظر بگیرید و جدول فاصله^۲ را برای گره c نمایش دهید.
- (ب) (۴ نمره) مشکل شمارش تا بی‌نهایت^۳ را در مسیریابی بردار فاصله^۴ در نظر بگیرید. آیا اگر هزینه یک لینک را کاهش دهیم، مشکل شمارش تا بی‌نهایت رخ خواهد داد؟ چرا؟ اگر دو گره که در حال حاضر پیوندی بینشان وجود ندارد را به هم متصل کنیم چطور؟
- (ج) (۴ نمره) ویژگی‌های یک الگوریتم مسیریابی متمرکز و توزیع شده را مقایسه و تفاوت آن‌ها را شرح دهید و برای هر یک مثالی از یک پروتکل مسیریابی بیاورید.
۲. (۱۰ نمره) شبکه نشان داده شده زیر را در نظر بگیرید. با استفاده از الگوریتم دایکسترا^۵، کوتاه‌ترین مسیر از هر یک از گره‌های مبدأ مشخص شده زیر را به سایر گره‌های شبکه محاسبه کنید. نتایج و مراحل کار خود را در قالب جدولی ارائه دهید.



- (آ) (۵ نمره) کوتاه‌ترین مسیر از گره t به سایر گره‌ها
- (ب) (۵ نمره) کوتاه‌ترین مسیر از گره u به سایر گره‌ها

^۱ Distance-Vector
^۲ Distance Table
^۳ count-to-infinity
^۴ distance vector routing
^۵ Dijkstra

۳. (۱۵ نمره) باتوجه به مسیریابی BGP، به سوالات زیر پاسخ دهید:

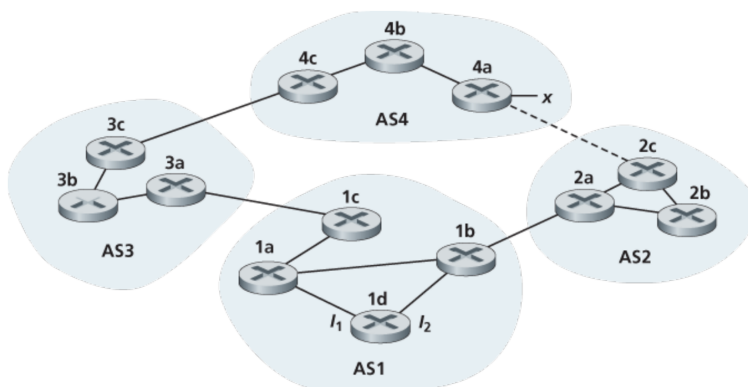
(آ) (۵ نمره) دلایل استفاده از پروتکل‌های مسیریابی مختلف بین ناحیه‌ای^۶ و درون ناحیه‌ای^۷ در اینترنت را شرح دهید.

(ب) (۴ نمره) آیا یک روتر BGP همیشه مسیری بدون حلقه با کوتاه‌ترین طول AS-path را انتخاب خواهد کرد؟ پاسخ خود را توجیه کنید

(ج) (۳ نمره) با توجه به شکل زیر، فرض کنید AS^۳ و AS^۲ برای پروتکل مسیریابی درون‌ناحیه‌ای خود از OSPF استفاده می‌کنند. AS^۱ و AS^۴ برای پروتکل مسیریابی درون‌شبکه‌ای خود از RIP استفاده می‌کنند و همچنین از eBGP و iBGP برای پروتکل مسیریابی بین شبکه‌ای استفاده می‌شود. در ابتدا هیچ لینک فیزیکی بین AS^۲ و AS^۴ وجود ندارد.

روتر ۳c و ۳a به ترتیب، پیشوند x^۸ را از کدام پروتکل مسیریابی (OSPF, RIP, eBGP, iBGP) یاد می‌گیرند؟

(د) (۳ نمره) هنگامی که مسیریاب ۱d از x مطلع می‌شود، یک ورودی (x, I) در جدول ارسال خود قرار خواهد داد. حال فرض کنید که یک لینک فیزیکی بین AS^۲ و AS^۴ وجود دارد که با خط نقطه‌چین نشان داده شده است و همچنین روتر ۱d متوجه می‌شود که x از طریق AS^۲ و همچنین از طریق AS^۳ قابل دسترسی است. آیا I روی I₁ تنظیم می‌شود یا I₂؟ دلیل خود را در یک جمله توضیح دهید.



۴. (۲۰ نمره) با توجه به مفاهیم SDN به سوالات زیر پاسخ دهید.

(آ) (۴ نمره) ساختار یک ورودی جریان^۹ در پروتکل OpenFlow را شرح دهید.

اجزای اصلی مانند Match Fields، Instructions و Counters را توضیح داده و نقش هر یک در اعمال سیاست‌های شبکه را به طور خلاصه بیان کنید.

(ب) (۳ نمره) نقش اصلی لایه ارتباطات^{۱۰}، لایه مدیریت وضعیت گسترده شبکه^{۱۱} و همچنین لایه برنامه کاربردی کنترل شبکه^{۱۲} در یک کنترل‌کننده SDN را شرح دهید.

(ج) (۵ نمره) توضیح دهید که چگونه کنترلر SDN با استفاده از پروتکل OpenFlow با سوئیچ‌های شبکه (صفحه داده) تعامل می‌کند تا این سیاست امنیتی را پیاده‌سازی کند. در پاسخ خود، مراحل کلی فرآیند از لحظه ورود اولین بسته مربوط به یک جریان جدید به سوئیچ تا اعمال سیاست توسط کنترلر را شرح دهید و به نقش پیام‌های کلیدی OpenFlow مانند PacketIn، FlowMod، و PacketOut (در صورت لزوم) اشاره کنید.

inter-AS^۶

intra-AS^۷

prefix^۸

Flow Entry^۹

communication layer^{۱۰}

network-wide state-management layer^{۱۱}

network-control application layer^{۱۲}

- (د) (۳ نمره) هدف لایه انتزاع سرویس^{۱۳} در کنترلر شبکه‌ی نرم‌افزارمحور OpenDaylight چیست؟
- (ه) (۵ نمره) فرض کنید در یک شبکه مبتنی بر SDN، یک لینک دچار خرابی شود. توضیح دهید که چه تعاملات بین کنترلر SDN، سوئیچ‌ها و برنامه‌های کنترلر شبکه انجام می‌شود تا مسیرهای جدید ایجاد شده و شبکه به حالت پایدار بازگردد. نقش هر یک از این اجزا را در این فرآیند شرح دهید

^{۱۳} service abstraction layer

سوالات عملی

تحويل دادنی:

خروجی مورد انتظار سوالات عملی شامل تصاویر مراحل اجرایی و نتایج بدست آمده و همچنین پاسخ به سوالات و ارسال کدهای خواسته شده است.

۵. (۲۰ نمره) آشنایی با OpenDaylight و Mininet

این تمرین جهت آشنایی با مفاهیم شبکه‌های نرم‌افزارمحور^{۱۴} با استفاده از ابزارهای OpenDaylight و Mininet است. هدف این تمرین، یادگیری عملی و آشنایی با دستورات کلیدی این دو ابزار و همچنین درک پارامترها و مفاهیم پایه SDN است.

اهداف تمرین:

- آشنایی با Mininet به عنوان یک شبیه‌ساز شبکه برای SDN.
- یادگیری نحوه ساخت توپولوژی‌های شبکه ساده با استفاده از Mininet.
- آشنایی با OpenDaylight به عنوان یک کنترلر SDN متن‌باز.
- نصب و راه‌اندازی OpenDaylight و اتصال آن به Mininet.
- مشاهده و مدیریت توپولوژی شبکه از طریق واسط کاربری OpenDaylight.
- تحقیق و بررسی پارامترهای مختلف در دستورات و تنظیمات OpenDaylight و Mininet.

Mininet امکان ایجاد سریع توپولوژی‌های مجازی شامل میزبان‌ها، سوئیچ‌ها، کنترلرها و لینک‌ها را فراهم می‌کند. در این بخش، با نحوه نصب Mininet و اجرای یک مثال ساده آشنا می‌شویم.

گام ۱: نصب و راه‌اندازی Mininet

می‌توانید از طریق مدیر بسته توزیع خود سرویس Mininet را نصب کنید (برای مثال `sudo apt install mininet` در اوبونتو). برای جزئیات بیشتر به مستندات رسمی Mininet مراجعه شود.

آزمایش نصب:

- پس از نصب دستور زیر را اجرا کرده و خروجی آن را تحلیل کنید:

```
sudo mn --test pingall
```

- در ترمینال دیگر از دستور زیر برای مشاهده جداول جریان سوئیچ s1 استفاده کنید. خروجی این دستور را تفسیر کنید.

```
sudo ovs-ofctl dump-flows s1 -O OpenFlow13
```

گام ۲: ایجاد توپولوژی سفارشی در Mininet

- یک توپولوژی خطی با ۳ سوئیچ ایجاد کنید و دستورات پایه net و dump و nodes و links را در mininet اجرا کنید و خروجی را تحلیل کنید.
- حال یک توپولوژی درختی با depth=2 و fanout=2 ایجاد کنید. تحلیل خود را از توپولوژی ایجاد شده بیان کنید.
- پارامترهای مختلف دستور mn مانند topo، switch، controller، mac به‌طور خلاصه تحلیل کنید. چه نوع توپولوژی‌های پیش‌فرض دیگری در Mininet وجود دارد؟

گام ۳: نصب و راه‌اندازی OpenDaylight

- ابزار OpenDaylight را نصب کنید و مراحل نصب را در گزارش بیاورید.
- واسط کاربری DLUX ابزاری بصری برای مشاهده توپولوژی، گره‌ها، جریان‌ها و وضعیت کنترلر است. از طریق مرورگر به واسط کاربری DLUX متصل شوید.
- راهنمایی: دسترسی به واسط کاربری معمولاً از طریق پورت ۸۱۸۱ مانند نشانی زیر صورت می‌گیرد و نام کاربری و رمز عبور پیش‌فرض (admin/admin) است.

```
http://<IP-Address>:8181/index.html
```

تصاویر مراحل خواسته شده، در گزارش ارائه شود.

گام ۴: اتصال به Mininet و تحلیل توپولوژی شبکه در DLUX

- دستور زیر را اجرا کنید و آن را تحلیل کنید.

```
sudo mn --controller=remote,ip=<ODL_CONTROLLER_IP>,port=6653 --
      topo single,3 --switch ovs,protocols=OpenFlow13
```

توجه داشته باشید که ODL_CONTROLLER_IP را باید با آدرس IP واقعی کنترلر OpenDaylight خود جایگزین کنید.

- پس از اجرای دستور، از طریق رابط کاربری OpenDaylight در مرورگر خود تصویر Topology و Nodes را ارائه کنید.

گام ۵: طراحی و پیاده‌سازی توپولوژی مورد نظر با اسکریپت پایتون

تاکنون با اجرای دستورات Mininet و ایجاد توپولوژی‌های ساده شبکه آشنا شده‌اید. حال توپولوژی شبکه دلخواه خود را با استفاده از اسکریپت پایتون تعریف کنید.

- با استفاده از یک کد پایتون یک توپولوژی با ۴ هاست و ۳ سوئیچ ایجاد کنید.

```
#!/usr/bin/env python

from mininet.topo import Topo

class MyTopo( Topo ):

    def build( self ):
        .
        .
        .

topos = { 'mytopo': ( lambda: MyTopo() ) }
```

- پس از ایجاد کد پایتون خود دستور زیر را اجرا کنید :

```
sudo mn --custom mytopo.py --topo mytopo --controller=remote,ip=<
    ODL_CONTROLLER_IP>,port=6653 --switch ovsk,protocols=OpenFlow13
```

- دستور pingall را اجرا کنید؛ انتظار می‌رود تمام هاست‌ها یکدیگر را با موفقیت پینگ کنند. خروجی اجرایی در گزارش آورده شود.
- پس از اجرای دستور، از طریق رابط کاربری OpenDaylight در مرورگر خود تصویر Topology و Nodes را ارائه کنید.

۶. (۲۰ نمره) پیاده‌سازی سرویس‌های مبتنی بر پروتکل ICMP

پروتکل پیام کنترل اینترنت^{۱۵} یک جزء اساسی از مجموعه پروتکل‌های اینترنت است که برای اهداف تشخیصی و گزارش خطا در شبکه استفاده می‌شود. ابزارهای شبکه مانند Ping و Traceroute به طور گسترده از پیام‌های ICMP برای بررسی قابلیت دسترسی میزبان‌ها، اندازه‌گیری تأخیر شبکه و کشف مسیرهای شبکه استفاده می‌کنند.

بخش اول: تحقیق و مبانی نظری ICMP

- در مورد پروتکل ICMP به ویژه پیام‌های Echo Request، Echo Reply و همچنین انواع Code و Type های مربوط به این دو پیام تحقیق کنید.

^{۱۵}Internet Control Message Protocol (ICMP)

بخش دوم: پیاده‌سازی عملیاتی

یک برنامه کاربردی جهت درخواست اکو ICMP در پایتون بنویسید. این برنامه باید قادر به انجام موارد زیر باشد:

- ارسال بسته‌های ICMP Echo Request به یک میزبان مقصد مشخص شده توسط کاربر.
- دریافت و پردازش صحیح بسته‌های ICMP Echo Reply از میزبان مقصد.
- امکان مشخص کردن تعداد بسته‌های Echo Request برای ارسال (مثلاً با یک پارامتر ورودی).
- پس از اتمام ارسال تمام بسته‌ها، خلاصه‌ای از آمار پینگ شامل تعداد کل بسته‌های ارسالی، تعداد بسته‌های دریافتی، درصد بسته‌های گم‌شده، و حداقل و حداکثر و میانگین RTT برای بسته‌های موفق نمایش دهید.
- پس از نوشتن کد مورد نظر، یک درخواست اکو به یک آدرس IP معتبر و یک درخواست اکو به یک آدرس IP نامعتبر ارسال کنید.
- پیام خطای ICMP ایجادشده را با Wireshark ضبط کرده و در گزارش خود بیاورید و سپس تحلیل کوتاهی در مورد نوع و کد پیام خطای ICMP داشته باشید.

بخش سوم: پیاده‌سازی برنامه Traceroute

Traceroute یک ابزار تشخیص شبکه است که مسیر ارتباطی بین مبدا و مقصد را با شناسایی میزبان‌های میانی تحلیل می‌کند.

برنامه‌ای به زبان پایتون بنویسید که عملکرد Traceroute را با استفاده از پروتکل ICMP شبیه‌سازی کند. برنامه باید:

- یک میزبان مقصد را به عنوان ورودی دریافت کند.
 - بسته‌های ICMP Echo Request با TTL افزایشی ارسال کند.
 - پاسخ‌های ICMP Time Exceeded از روترهای میانی و ICMP Echo Reply از مقصد را پردازش کند.
- خروجی برنامه: اطلاعات هر هاپ شامل شماره و زمان‌های رفت و همچنین برگشت پیام ارسالی را نمایش دهد.
- برنامه Traceroute خود را برای ردیابی مسیر به یک مقصد اینترنتی اجرا کنید. خروجی کامل اجرای برنامه را در گزارش خود درج نمایید.
 - توضیح دهید که چگونه پیام ICMP Time Exceeded و اطلاعات موجود در آن (به ویژه بسته IP اصلی که سبب بروز خطا شده است) به شناسایی روترهای میانی کمک می‌کند.
 - در خروجی Traceroute معمولاً هاپ‌هایی با علامت * مشخص شده‌اند، دلایل احتمالی بروز این وضعیت را شرح دهید. آیا این مسئله همواره به معنای وجود مشکل در شبکه است؟