



تمرین برنامه‌نویسی دوم^۱ شبکه‌های کامپیوتری

بهار ۱۳۸۹

مدرس: مهدی خرازی

Algorhyme^۲

“ I think that I shall never see
A graph more lovely than a tree

A tree whose crucial property
Is loop-free connectivity

A tree which must be sure to span
So packets can reach every LAN

First the Root must be selected
By ID it is elected

Least cost paths from Root are traced.
In the tree these paths are placed

A mesh is made by folks Like me
Then bridges find a spanning tree^۳

۱ با تشکر از صادق دری، شایان پویا، بهنام مومنی، حسن اسلامی و امیر شیخ‌ها

2 By: Radia Perlman

مقدمه:

همان‌طور که در کلاس بیان شد، سویچ‌ها ابزارهایی برای مسیریابی بسته‌ها در لایه‌ی دو هستند. مهمترین مزیت سویچ‌ها نسبت به ابزارهایی مانند هاب^۳ (که پیش از سویچ وجود داشتند) این است که سویچ‌ها بسته‌های دریافتی را تنها بر روی interface ای که مقصد در آن وجود دارد می‌فرستند.

سویچ‌ها با دریافت هر بسته، آدرس ماشین فرستنده و interface ای که بسته از آن به سویچ رسیده است را ذخیره می‌کنند و به این شکل در صورتی که در آینده بسته‌ای به مقصد این ماشین به سویچ رسید، سویچ می‌داند که باید بسته را فقط به همان interface بفرستد. این اطلاعات معمولاً در جدول‌هایی درون سویچ‌ها ذخیره می‌شوند.

همچنین، در کلاس بیان شد که سویچ‌ها برای ارسال صحیح داده‌ها و جلوگیری از به وجود آمدن حلقه از پروتکل تشکیل درخت فراگیر STP^۴ استفاده می‌کنند. به این شکل، بعد از روشن شدن سیستم، سویچ‌ها باید پیام‌های STP رد و بدل کنند و درخت فراگیر کمینه را بسازند و سپس قابلیت مسیریابی بسته‌ها را دارا خواهند بود.

در این تمرین شما باید برنامه‌ای برای کنترل کردن چنین سویچ‌هایی بنویسید. یعنی پروتکل STP و قابلیت learning را روی سویچ‌ها پیاده‌سازی کنید.

محیط:

در این تمرین از محیط پرتو^۵ برای اجرای برنامه‌ی شما استفاده خواهد شد. پرتو قابلیت ایجاد محیطی مجازی دارد که در آن می‌توانید توپولوژی‌های گوناگونی داشته باشید. برای کار با این محیط لازم است که برنامه‌ی شما بر روی «چارچوب کاربر» پرتو پیاده‌سازی شود. برای اطلاعات بیشتر در مورد این چارچوب به مستند «راهنمای چارچوب کاربر» مراجعه کنید.

توپولوژی:

توپولوژی اولیه که در اختیار هر دانشجو قرار می‌گیرد مانند شکل ۱^۶ است.

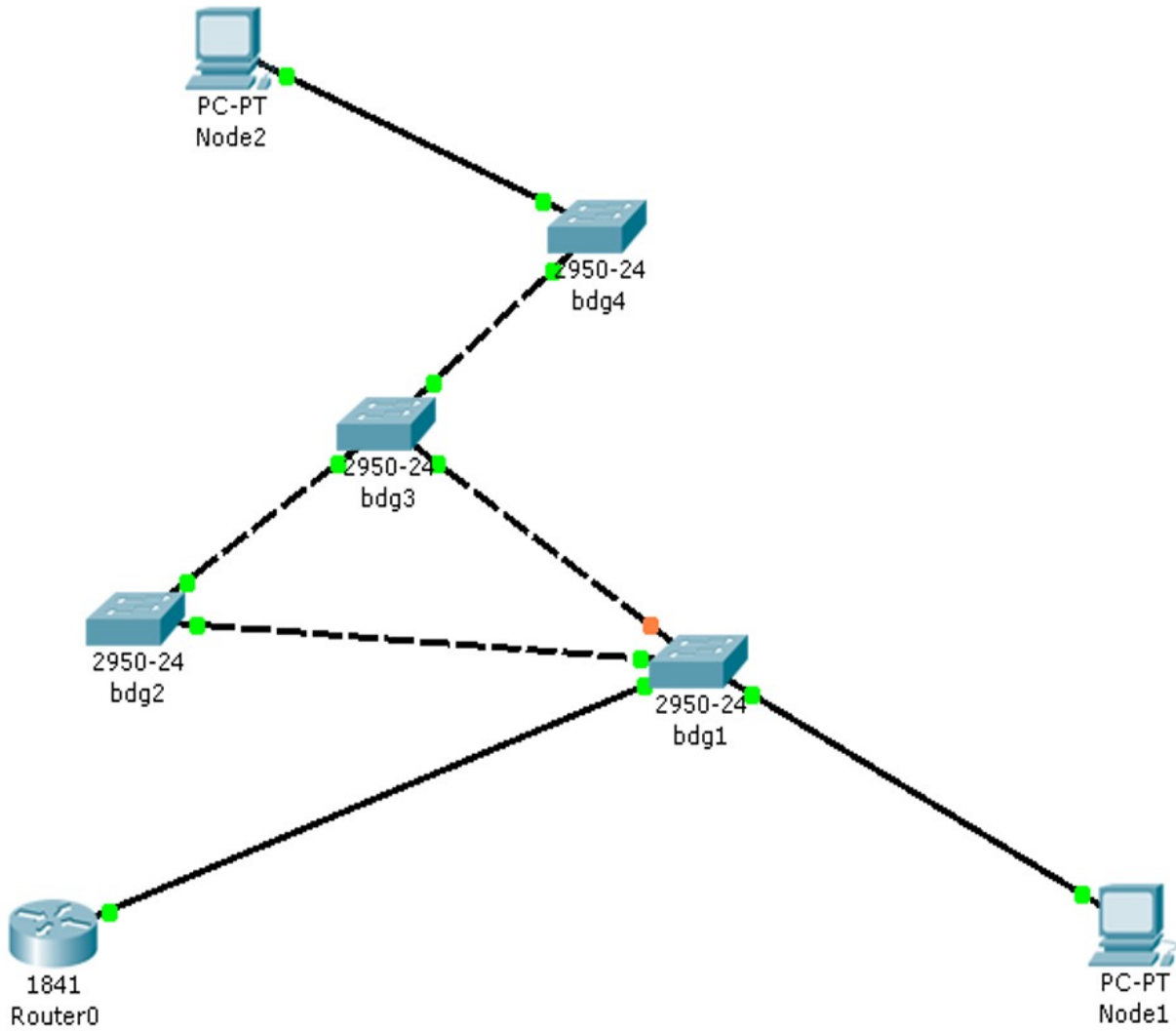
در این شکل رنگ سبز به معنای فعال بودن لینک و قرمز به معنای غیر فعال بودن آن است. از برنامه‌ی شما انتظار می‌رود که با قرار گرفتن روی چهار سویچ شکل زیر، امکان دسترسی از محیط بیرون به host های مختلف را بدهد. برنامه‌ی شما به هیچ وجه نباید وابسته به این توپولوژی باشد و یا فرض خاصی در مورد توپولوژی انجام دهد.

3 Hub

4 Spanning Tree Protocol

5 Portable And Reliable Tool fOr Virtualization (PARTOV)

۶ این شکل با استفاده از نرم‌افزار Packet Tracer رسم شده است.



شکل ۱: توپولوژی نمونه برای نحوه‌ی اتصال سویچ‌ها

نکات:

- همان‌طور که اشاره شد، برنامه‌ی شما باید روی هر توپولوژی‌ای کار کند. برای تست کردن برنامه‌ی شما از توپولوژی‌های دیگری نیز استفاده خواهد شد. به ویژه توجه کنید که کدی که روی این چهار سویچ اجرا می‌کنید دقیقاً یکسان‌اند و نباید با استفاده از نام آن‌ها کد خاصی برای آن‌ها اجرا کنید.^۷
- نام این توپولوژی **bridge** است و نام سویچ‌هایی که شما باید برنامه‌تان را روی آن‌ها اجرا کنید عبارتند از:
 - bdg1
 - bdg2
 - bdg3
 - bdg4

^۷ مثلاً وجود شرطی با مفهوم «اگر سویچ یک هستیم» غیر قابل قبول است.

- در صورت از کار افتادن یکی از سویچ‌ها یا اضافه شدن یک سویچ جدید درخت فراگیر را بازسازی کند و بعد از مدت معقولی (۱۰ ثانیه برای چهار سویچ) به حالت پایدار برسد.
- سویچ‌ها باید به صورت هوشمند عمل کنند و در صورت امکان^۸، بسته را فقط بر روی interface ای که ماشین مقصد در آن قرار دارد بفرستند.
- جدول‌های learn شده باید به صورت دوره‌ای به روز رسانی شوند. همان‌طور که در کلاس بیان شد، در صورتی که یک ماشین را از یک interface یک سویچ قطع کنیم و به interface دیگر آن اضافه کنیم، بعد از مدت معقولی (۳۰ ثانیه) سویچ باید بفهمد که interface قبلی که در جدولش نگهداری می‌شود غلط است و آن را به روز کند. این به‌روز رسانی به این شکل انجام می‌شود که بعد از مدتی که هیچ بسته‌ای از ماشین مورد نظر دریافت نشد، سویچ entry مربوط به آن ماشین را از جدولش پاک می‌کند و بعد از آن در صورتی که پیام جدیدی به مقصد این ماشین دریافت کرد، روی همه‌ی interface‌هایش می‌فرستد، تا وقتی که بسته‌ی جدید از این ماشین دریافت کند و دوباره یک entry برای آن به وجود آورد.
- بعد از اجرای برنامه‌ی شما، انتظار می‌رود که سویچ‌ها بعد از setup اولیه، قابلیت مسیریابی بسته‌ها را داشته باشند. برای تست این قابلیت دو ماشین نمایش داده شده در توپولوژی ping می‌شوند. پیام‌های ICMP حاصل از ping باید از طریق این سویچ‌ها به ماشین مورد نظر برسد و جواب‌های ICMP نیز به درستی مسیر بازگشت را بیابند.
- با توجه به پیچیده بودن الگوریتم و بسته‌های پروتکل فعلی STP که Rapid STP^۹ نام دارد. شما باید نسخه‌ی قدیمی پروتکل STP را پیاده‌سازی کنید. این نسخه همان الگوریتمی است که در کلاس با آن آشنا شدید. می‌توانید header دلخواه خود را برای این پروتکل طراحی کنید و از header استاندارد استفاده نکنید. اما در هر صورت، پروتکل شما باید مطابق مطالب درس‌داده شده در کلاس کار کند.
- برای اطلاعات بیشتر و دقیق‌تر در مورد STP به مراجع زیر مراجعه کنید.
 ۱. http://en.wikipedia.org/wiki/Spanning_tree_protocol
 ۲. <http://www.informit.com/content/images/1587201429/samplechapter/1587201429content.pdf>
 ۳. <http://www1.cs.columbia.edu/~ji/F02/ir02/p44-perlman.pdf>
- برای تست کردن برنامه‌تان از سناریوی زیر استفاده کنید. در کل زمان اجرای این سناریو، بسته‌های ping باید به صورت مداوم فرستاده شوند.

«چهار نسخه از برنامه‌تان را روی چهار سویچ شکل ۱ اجرا کنید. بعد از کامل شدن درخت STP، سویچ bdg2 را قطع کنید.^{۱۰} بعد از تشکیل درخت جدید روی سه سویچ باقی‌مانده، دوباره برنامه‌تان را روی bdg2 اجرا کنید. درخت فراگیر باید دوباره به شکل اولیه برگردد.»

تذکر:

- از فرستادن کل یا قسمتی از جواب تمرین به گروه پستی جداً خودداری کنید.
- فرستادن کل یا قسمتی از برنامه‌تان برای افراد دیگر، یا استفاده از کل یا قسمتی از برنامه‌ی فردی دیگر به نام خود، تقلب محسوب می‌شود.
- از فرستادن فایل‌های اجرایی خودداری کنید و تنها کدهایتان را بفرستید.^{۱۱}

۸ اگر ماشین فوق حداقل یک بسته فرستاده باشد که از این سویچ گذشته باشد.

9 RSTP

۱۰ مثلاً با زدن `ctrl + C` در ترمینالی که یک نسخه از برنامه‌تان را اجرا می‌کنید. با این کار، خاموش شدن سویچ bdg2 را شبیه‌سازی می‌کنید.

۱۱ بعد از اتمام نوشتن برنامه عبارت `make clean` را در ترمینال وارد کنید.

- بعد از اتمام تمرین، پوشه‌ی کاری خود را به pa2-80123456 تغییر نام دهید که در آن به جای ۸۰۱۲۳۴۵۶ شماره‌ی دانشجویی خود را بیاورید. این پوشه را فشرده کنید و به آدرس‌های *kharrazi@sharif.edu* و *dorri@ce.sharif.edu* بفرستید. موضوع^{۱۲} این پست الکترونیکی را نیز دقیقاً همین عبارت قرار دهید.