



این مستند به ضمیمه تمرین شماره ۰ قرار گرفته است و نحوه کار با سامانه‌ی داوری<sup>۱</sup> (که در وب‌گاه درس قرار گرفته) و نحوه نمرده‌دهی را به شما توضیح می‌دهد.

## ۱. نحوه کار با سامانه‌ی داوری

داوری این تمرین همانند داوری سایر تمرینات است، لذا توصیه می‌شود کار با این سامانه را در همین تمرین فراگیرید. برای کار با سامانه‌ی داوری ابتدا پرونده‌ی آرشیو را از وب‌گاه درس بارگیری کنید و از حالت zip خارج کنید، سپس وارد پوشه‌ی اصلی آن شوید و مراحل زیر را دنبال کنید.

### ۱.۱. تنظیمات پیش فرض

ابتدا باید کد خود را در پوشه `script-runners/Tester/codes` بدون هیچ پوشه واسطی قرار دهید، یعنی مثلاً `new.sh` شما باید در آدرس `script-runners/Tester/codes/new.sh` قرار گرفته باشد. نکته: دقت کنید که از قبل کد درست را در این قسمت قرار داده‌ایم تا در صورتی که علاقه‌مند بودید سامانه‌ی داوری را با کد درست اجرا کنید و از صحت اجرای درست این سامانه روی ماشین خود مطمئن شوید. یعنی اگر کد خود را در این قسمت قرار ندهید و ادامه مراحل گفته شده را دنبال کنید، پس از اجرای سامانه‌ی داوری باید تمام نمرات شما برابر ۱۰۰ شود. پس در صورتی که از صحت سامانه مطمئن شدید کد خود را در این قسمت قرار دهید و کدی که وجود دارد را پاک کنید تا داوری روی کد شما اجرا شود.

کد خود را با دستور `make` کامپایل کنید تا پرونده‌ی اجرایی آن ساخته شود، سپس پرونده‌ی `info.sh` درون کدتان را ویرایش کرده و قسمت `map` را برابر با `DHCP_Simple` قرار دهید. همین کار را برای `info.sh`

\* با سپاس از سولماز سلیمی، امیرپاشا قابوسی، آرش ملاجعفری و مهدی بهروزی‌خواه

<sup>۱</sup>Judge

کد درست که درون پوشه‌ی `script-runners/Tester/judge` قرار دارد انجام دهید و با همان اطلاعات کاربری خودتان پر کنید. دقت کنید محتویات این پرونده دقیقاً باید شبیه به پرونده‌ای باشد که در پوشه کد خودتان قرار دارد.

سپس آدرس دقیق محل پوشه `codes` را باید درون پرونده‌ی `script-runners/Tester/config.json` در قسمت `cf_path` قرار دهید. همچنین قسمت `judge_path` را نیز با آدرس مطلق پوشه‌ی `judge` که در آدرس `script-runners/Tester/judge` برای شما قرار داده شده است را پر کنید. عدد `sleep_time` مشخص می‌کند پس از اجرای هر دستور روی هر گره چند ثانیه صبر کند تا دستور بعدی را به گره دیگری وارد کند. در صورتی که ارتباط اینترنتی خوبی دارید و برای ارتباط با سامانه‌ی پرتو مشکلی ندارید به این عدد دست نزنید. در صورتی که اینترنت شما کند است یا ارتباط شما با سامانه‌ی پرتو دچار مشکل است، ممکن است در هنگام اجرای `judge` به مشکل برخوردید و بسته‌های شما قبل از رسیدن داوری شوند. در نتیجه در این حالت‌ها بهتر است این عدد را تا ۱۰ افزایش دهید (افزایش این عدد باعث کندتر اجرای سامانه‌ی داوری خواهد شد و شما باید زمان بیشتری را منتظر بمانید تا نمره خود را ببینید!).

## ۲.۱. آماده سازی

برای اجرای کد `judge` شما نیاز به `virtualenv`، `tcpdump`، `Python 2.7` دارید. پس از نصب این بسته‌ها، درون پوشه‌ای که از حالت فشرده خارج کردید یک ترمینال باز کنید و پرونده‌ی `init.sh` را اجرا کنید. برای این‌کار می‌توانید از دستور: `./init.sh` استفاده کنید. دقت کنید که باید قبل از اجرای این دستور به اینترنت وصل باشید. پس از این دستور باید `Python 2.7` و تمام پیش‌نیازهای اجرایی `judge` برای شما درون پوشه `venv` قرار گرفته باشد.

حال دستور `source start.sh` را اجرا کنید تا از این به بعد ترمینالی که باز کرده‌اید از پایتون ساخته شده استفاده کند. پس تا وقتی که ترمینال خود را نبسته‌اید، می‌توانید از آن استفاده کنید. دقت کنید که بار هر بار بستن ترمینال خود و باز کردن آن به منظور اجرای سامانه‌ی داوری، باید حتماً همین دستور را اجرا کنید تا پیش‌نیازهای پایتون بارگذاری شود.

حال باید با دستور `cd script-runners` به پوشه `script-runner` بروید. تا همه چیز برای اجرای کد محیا شود. برای اجرا کافی است دستور `./run.sh` را بزنید تا اجرای کد سامانه‌ی داوری شروع شود.

## ۳.۱. خروجی‌های جاج

هنگامی که سامانه‌ی داوری در حال اجرا است، به شما گزارش لحظه‌ای می‌دهد که در حال چه کاری است. هر بار به سراغ یک آزمون می‌رود، اسم آزمون را به رنگ آبی چاپ می‌کند و شروع به اجرای سناریوهای مختلف می‌کند، و

می‌توانید ببینید که چه بسته‌هایی رد و بدل می‌شوند. در انتهای هر سناریو بسته‌ها و خروجی‌های شما بررسی می‌شوند و اگر نمره آن سناریو را گرفتید، نمره شما به‌روز می‌شود. در نهایت برای هر آزمون جمع نمره تمام سناریوهای آن‌ها برابر ۱۰۰ می‌شود. پس اگر کد شما تمام تست‌ها را با موفقیت بگذراند باید در انتها ۴ عدد ۱۰۰ ببینید.

برای این‌که بفهمید چه بسته‌هایی رد و بدل شده و هر گره‌ای چه چیزی چاپ کرده پس از هر بار اجرای سامانه‌ی داوری، یک پوشه با نام `log` برای شما ساخته می‌شود که درون آن به ازای هر سناریو یک پوشه با نام آن سناریو قرار دارد. یک نمونه از `log` درست نیز برای شما با نام `correct_log` قرار داده شده است که در آخر باید `log` شما نیز مانند نمونه‌ی صحیح این شود. درون هر سناریو خروجی تمام گره‌های درگیر آن تست را برای شما قرار داده‌ایم. در پرونده‌های `stdout` خروجی‌ای است که گره مورد نظر چاپ کرده است. پرونده‌های `recv` بسته‌های دریافتی هر گره روی `interface` مربوطه‌اش است. و پرونده‌های `send` بسته‌های ارسالی هر گره می‌باشند.

برای باز کردن پرونده‌های `pcap`. نرم افزارهای مختلفی وجود دارد، از جمله این نرم‌افزارها می‌توانید از نرم افزار معروف `wireshark` استفاده کنید تا هر بسته شما را به تفکیک سرآیندهایش نشان دهد.

پرونده‌ی `error.log` نیز در صورتی که مشکلی نداشته باشید، خالی است و در صورتی که در آن سناریو مشکلی داشته باشید، مشکل شما را می‌نویسد.

#### ۴.۱. اطلاعات اضافی

در صورتی که راجع به نمره دهی جزئی‌تر هر سناریو و سامانه‌ی داوری اطلاعات بیشتری می‌خواهید، می‌توانید به پرونده‌های مربوط به هر آزمون مراجعه کنید. برای اینکار کافی است، به پوشه‌ی `script-runners/Tester/tests/` مراجعه کنید. در اینجا پرونده‌ی هر آزمون را می‌بینید و درون هر آزمون می‌توانید هر سناریو را ببینید. در انتهای هر سناریو، خروجی‌هایی که بررسی می‌شوند را می‌توانید ببینید. اطلاعات بیشتری در مورد آماده سازی جاج در فایل `Readme` نیز قرار گرفته است که می‌توانید آن بخش را نیز مطالعه کنید.

## ۲. موارد آزمون

همانطور که در صورت تمرین گفته شده بود، حالت‌های مختلفی برای مسئله وجود دارد. برای صحت عملکرد کد شما، آزمون‌های مختلفی در قالب سناریوهای متفاوتی طرح شده‌اند. در تمام این سناریوها از نقشه `DHCP_Simple` استفاده شده است. دقت کنید که نقشه آزمون نهایی با این نقشه متفاوت خواهد بود.

هر آزمون شامل چند سناریو است. در هر آزمون تنها تعداد خاصی از اعضای نقشه فعال هستند. هر گره، دو حالت دارد که یا کد درست روی آن در حال اجراست، یا کد شما. برای تمام سناریوهای یک آزمون، این اعضا و حالتشان ثابت است، اما در اول هر سناریو این اعضا ریست می‌شوند تا عملکرد گره‌های شما در سناریو قبلی تأثیری در سناریو جدید نگذارد. هر سناریو به این صورت است که دستوراتی به اعضای شبکه وارد می‌شود تا بسته‌هایی رد و بدل شوند. سپس تمام بسته‌های خروجی، ورودی و تمام پیام‌های چاپ شده هر گره مورد بررسی قرار می‌گیرند و درست بودن هر کدام قسمتی از نمره شما را تشکیل می‌دهد.

اعضای هر سناریو را با لیستی نشان می‌دهیم که حرف c در آن لیست یعنی در این گره کد شما اجرا می‌شود و در صورتی که حرف z باشد یعنی کد درست در آن گره اجرا می‌شود. همچنین در انتهای هر سناریو معلوم است که چه دستوراتی به هر گره وارد می‌شود و چه مواردی مورد آزمون است و هر بخش چند نمره دارد. در صورتی که می‌خواهید این اطلاعات را ببینید دو پرونده‌ی مربوط به آزمون را که در آدرس `/script-runners/Tester/tests` قرار دارد مطالعه کنید.

در این تمرین عملکرد دو ماشین را باید پیاده‌سازی کنید که هر کدام در مجموع 50 نمره دارند. جزییات هر بخش به این صورت است که هر کدام از مواردی که به صورت جدا برای هر ماشین گفته شده، مثلاً برای کارخواه، دریافت انواع بسته‌ها، پردازش دستورات و ارسال بسته‌ها تقریباً نمرات برابری دارند و مسیریابی و ترکیب حالت‌های پیچیده‌تر نمره بیشتری دارند. شما برای هر بخشی که با موفقیت کدنویسی کرده باشید، نمره خواهید گرفت. دقت کنید که مثلاً اگر شما نتوانید بسته `dhcp discover` درستی بفرستید، قطعاً فرآیند اختصاص آی‌پی آغاز نمی‌شود و در نتیجه نمرات بسته‌های دیگر را نیز نخواهید گرفت زیرا هیچ‌گاه نمی‌توان به آن مرحله رسید. اما اگر `dhcp discover` را درست زده باشید و قسمت‌های بعدی را زده باشید، نمره این بخش را خواهید گرفت.