



دانشکده‌ی مهندسی کامپیوتر

شبکه‌های کامپیوتری

مدرس: مهدی جعفری

## تمرین عملی لایه Network

گردآورندگان: محمدامین آریان - هیرید بهنام

شنبه ۲ دی ۱۴۰۲

در این سوال می‌توانید از هر زبان برنامه نویسی استفاده کنید.  
در این برنامه نیازی نیست که به چندین کلاینت همزمان سرویس دهید و می‌توانید همیشه فرض کنید که یک کلاینت وجود دارد.  
قسمت‌های مشخص شده با علامت \* سوالات و یا مطالب کاربردی هستند که جواب دادن و خواندن آنان اختیاری است اما نمره اضافی ندارد.

## ۱ Making UDP Great Again

می‌دانیم که پروتکل UDP یک پروتکل غیر قابل اطمینان است که نمی‌تواند رسیدن بسته به مقصد یا حفظ ترتیب بسته‌ها را تضمین کند. در این بخش باید یک پروتکل طراحی پیاده‌سازی کنید که قابل اطمینان باشد و ترتیب بسته‌ها را حفظ کند. برای این کار باید یک مسیر مطمئن با استفاده از پروتکل UDP طراحی کنید. می‌توانید از پروتکل‌های reliable data transfer (rdt) برای تضمین رسیدن بسته‌ها به مقصد و از روش‌هایی مانند Go-Back-N یا Selective Repeat یا هر روش دیگری که می‌خواهید استفاده کنید و می‌توانید از هر زبان برنامه‌نویسی‌ای استفاده کنید اما پیاده‌سازی شما باید بدون استفاده از کتابخانه‌های جانبی باشد. همچنین یک گزارش از پروتکل طراحی شده و نحوه پیاده‌سازی آن بنویسید و در کنار کد نوشته شده خود تحویل دهید.

## ۲ تحویل دادنی‌ها

فرمت فایل‌هایی که تحویل می‌دهید باید به صورت زیر باشد:

۱. فایل README که نام، نام‌خانوادگی، شماره دانشجویی و توضیحات کد و الگوریتمتان است.
۲. کدهای خود را می‌توانید به هر زبانی می‌خواهید بنویسید ولی پیشنهاد ما پایتون است.
۳. یک فایل Makefile که پروژه‌ی شما رو بیلد می‌کنه. در صورتی که از dependency خاصی استفاده می‌کنید حتما آن را در Makefile خود نصب کنید.

## ۳ تست و ارزیابی

برای تست خود می‌توانید از برنامه‌ی [Lossy Link](#) استفاده کنید. این برنامه برای شما یک ارتباط دو طرفه بر بستر UDP می‌سازد که هم برخی از پکت‌هایش گم می‌شود و هم اینکه برخی از پکت‌هایش به صورت جا به جا فرستاده می‌شود. پکت‌های بازگشتی فقط احتمال گم شدن دارند و هیچ گاه جا به جا نمی‌شوند.  
در ابتدا می‌توانید از قسمت [releases](#) برنامه‌ی متناسب به سیستم‌عامل و معماری خود را انتخاب کنید. سپس بعد از دانلود برنامه می‌توانید آن را بدین صورت اجرا کنید:

```
./lossy_link 127.0.0.1:12345 127.0.0.1:54321
```

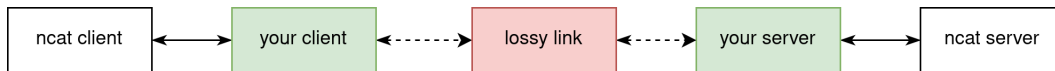
با این کار برنامه هر پکتی که بر روی پورت 12345 لوکال بیاید را برای پورت 54321 لوکال ارسال می‌کند.  
حال یک ncat بر روی پورت 54321 باز کنید به کمک دستور زیر:

```
ncat --recv-only -u -l 54321
```

و در نهایت به کمک دستور زیر دنباله‌ای از اعداد ۱ تا ۱۰۰۰ را برای `lossy_link` بفرستید.

```
seq 1000 | { while read; do sleep 0.01; echo "$REPLY"; done; } | ncat --send-only -u 127.0.0.1 12345
```

با این کار در ترمینالی که در قیل باز کرده بودید برخی از اعداد ۱ تا ۱۰۰۰ و گاهی با ترتیب تصادفی ظاهر می‌شوند. از برنامه‌ی شما انتظار داریم که در صورت نشستن بین `ncat server` و `ncat client` تمام مشکلات لینک `unreliable` را برطرف بکند و اعداد ۱ تا ۱۰۰۰ به ترتیب در ترمینال سرور ظاهر شوند. به صورت کلی معماری برنامه‌های تست به صورت زیر است:



موفق باشید