

پیاده سازی پروتکل ارسال بسته Go-Back-N به همراه تشخیص خطای CRC

در این پروژه قصد داریم ارسال بسته و error detection را بین یک فرستنده و یک گیرنده شبیه سازی کنیم.

برای این کار، فرض کنید دو سمت فرستنده و گیرنده وجود دارد و فرستنده قصد ارسال پیام هایی به گیرنده دارد. برای ارسال بسته ها پروتکل Go-back-N، و برای تشخیص خطا پروتکل CRC را پیاده سازی کنید. باید در سمت فرستنده با مشخص بود مقدار P یا همان چند جمله ای مولد، با روش Cyclic Redundancy Check ابتدا باید FCS یا همان codeword را ایجاد کنید و سپس به پیام اضافه کنید. پس از آن بسته ها را با استفاده از پروتکل Go-back-N، با اندازه پنجره دلخواه ارسال کنید.

از طرفی گیرنده باید قابلیت تشخیص گم شدن پکت های ارسالی از طرف فرستنده را داشته باشد و در صورت وجود خطا در فرایند ارسال، پاسخ مناسب (ACK,....) را ارسال کند.

همچنین توجه داشته باشید که گم شدن بسته ها، هم میتواند برای بسته های ارسالی از سمت فرستنده و هم برای پاسخ های ارسالی از گیرنده رخ دهد پس گم شدن بسته ها از هر سمت را با احتمال دلخواهی شبیه سازی کنید. (در فرستنده و گیرنده باید قابلیت تشخیص گم شدن بسته های ارسالی از سمت خودشان (مکانیزم timeout) را داشته باشد.)

در نظر داشته باشید که شما فقط باید رخ دادن خطا روی محتوای بسته های ارسالی (پیام فرستنده) از سمت فرستنده را در نظر بگیرید. (نیازی به پیاده سازی error checking برای پاسخ های ارسالی از گیرنده نیست و خطا بر محتوا فقط برای فرستنده رخ می دهد.)

در پیاده سازی پروژه به موارد زیر توجه داشته باشید:

- برقراری ارتباط بین فرستنده و گیرنده را در ترمینال نمایش دهید (قابلیت نمایش پیام های ارسالی در فرستنده و پیام های دریافتی در گیرنده)
- ابتدا ارسال بدون خطا را در نظر بگیرید و پیاده سازی کنید

- حال علاوه بر ارسال بدون خطا به طور رندوم تعدادی خطا در فرایند ارسال اضافه شود تا نحوه عملکرد پروتکل Go Back N نمایش داده شود (می تواند پس از نمایش پیام ها در ترمینال فرستنده یا قبل از نمایش در گیرنده باشد)
- مکانیزم time out را برای این پروتکل پیاده سازی کنید
- پروتکل CRC پیاده سازی شده باید قابلیت تغییر چند جمله مولد را دارا باشد
- تعدادی پیام های ارسالی از قبل آماده شده برای ارائه در نظر گرفته شود (در صورت نیاز حین ارائه این مقادیر با خواست TA ارائه گیرنده می تواند عوض شود)

بخش امتیازی :

- با استفاده از hamming code در فرستنده و گیرنده امکان تشخیص و تصحیح خطا را فراهم کنید.
- امکان استفاده از روش های دیگر برای تشخیص خطا از جمله checksum، parity check و 2D parity را اضافه کنید.
- با استفاده از Socket-programming در دو سمت فرایند ارسال را با تمام موارد اشاره شده پیاده سازی کنید.
- توضیحات : شما باید یک client و یک server با استفاده از Socket-programming تشکیل دهید. لینک های مربوطه ضمیمه شده است . همچنین ویدئو برای آشنایی با Socket-programming بارگذاری خواهد شد. (لازم به ذکر است در این فرایند سرور فقط گیرنده نیست و باید ack نیز ارسال کند. همچنین نوع کانکشن (TCP یا UDP) تفاوتی ندارد اما توصیه می شود که از TCP برای برقراری ارتباط استفاده کنید.)
- رابط کاربری (UI) قابل قبول برای نمایش فرایند ارسال داده پیاده کنید.

نکات تکمیلی :

- در صورت وجود ابهام یا اشکال در پیاده سازی در گروه مطرح کنید تا در سریع ترین زمان توسط یکی از TA ها پاسخ داده شود.
- ددلاین برای همه یکسان خواهد بود و فایل های ارسالی در VC باید بارگذاری شوند.
- پیشنهاد می شود ابتدا موارد اصلی پروژه پیاده سازی شوند و سپس در نسخه های بعدی موارد امتیازی اضافه شوند.
- برای دیتایی که قرار است ارسال شود، می توانید از یک آرایه از قبل مقدار دهی شده استفاده کنید.
- در هر مرحله چیزی که در client و server دریافت می شود را نمایش دهید.
- پیاده سازی با هر زبانی قابل قبول است اما پیشنهاد ما استفاده از پایتون است.
- در این پروژه گزارش نیست اما باید به طور کامل به کد و مفاهیم پروتکل های طرح شده مسلط باشید.

لینک های کمکی:

- socket programming: <https://realpython.com/python-sockets/>
- hamming code: <https://www.geeksforgeeks.org/hamming-code-in-computer-network/>

نکات ارسال

- کد خود را در فایل فشرده ای با اسمی به فرمت `Project3_Name_StudentId` ارسال نمایید.
- در صورتی که تقلب تشخیص داده شود، نمره پروژه ۱۰۰- خواهد شد.

موفق باشید