

**تکلیف کامپیوتری مبانی شبکه‌های بیسیم**

مدرس: دکتر سید حامد رستگار

دانشجو: محمدامین شهیدی نشرودکلی **-** ۹۶۵۲۲۱۷۷

دانشکده مهندسی کامپیوتر

پاییز ۱۴۰۰

**مقدمه**

پوشه ارسال شده شامل فایل مستندات با نام aloha-doc و همچنین یک فایل پایتون به نام aloha می‌باشد. در ادامه ابتدا به بررسی قسمت‌های مختلف کُد می‌پردازیم. پس از تشریح چگونگی کارکرد کُد، نتایج حاصل از اجرا برنامه برای موارد خواسته شده را بررسی می‌کنیم.

**تشریح کُد**

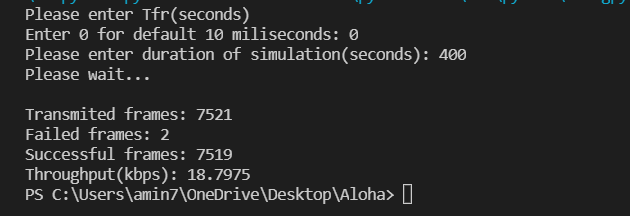
در این برنامه از قابلیت چند رشتگی پایتون استفاده شده‌ است. هر گره یک رشته است که سازوکار ALOHA را اجرا می‌کند.

از خط ۱ تا خط ۲۴ به ترتیب کتابخانه‌های مورد نیاز را افزوده‌ایم، سپس زمان ارسال هر فریم به ثانیه(Tfr) و مدت زمانی که می‌خواهیم شبیه سازی به طول انجامد را بر حسب ثانیه از کاربر گرفته‌ایم. اگر هنگام ورود Tfr کاربر عدد صفر را وارد کند، Tfr ده میلی ثانیه(که در صورت سوال داده شده) در نظر گرفته می‌شود. کاربر می‌تواند در ورودی‌ها عدد اعشاری نیز وارد نماید.

از خط ۲۸ تا ۸۳ تابعی تعریف شده است که مغز متفکر برنامه است. این تابع درواقع یک گره ALOHA است که عملیات ارسال را انجام می‌دهد. در خط ۲۹ تا ۳۴ متغیرهای برنامه را به تابع معرفی کرده‌ایم. در داخل حلقه اصلی برنامه یک گره یک مقدار نمایی تصادفی با میانگین ½ تولید می‌کند. از آنجایی که برای توضیع پواسن متناظر تعداد ارسال ها به صورت میانگین ۲ است، پس فاصله زمانی بین ۲ رخداد(۲ اقدام به ارسال) ½ ثانیه طول می‌کشد. بنابراین برای تولید مقدار تصادفی نمایی از ½ در خط ۳۹ استفاده شده است. سپس در خط ۴۲ گره به اندازه مقدار تصادفی تولید شده صبر می‌کند و سپس اقدام به ارسال می‌کند. متغیری به نام channelBusy داریم که نشان می‌دهد گرهی در حال ارسال است یا خیر. هر گره وقتی اقدام به ارسال کند، اگر channelBusy برابر با False باشد، آن را برابر با True می‌کند تا دیگر گره ها(همان رشته های پایتون) اشغال کانال را بفهمند. از طرفی اگر channelBusy برابر با True باشد و گرهی اقدام به ارسال کند، Collision رخ می‌دهد و فریم‌هایی که در آن لحظه در حال ارسال بوده اند از دست می‌روند. تعداد کل فریم های ارسال شده و از دست رفته در برنامه به کمک تمام گره‌ها نگهداری می‌شود و به کمک آنها در خط ۱۳۵ میزان Throughpu سیستم را محاسبه کرده‌ایم.

در خط ۸۶ تا ۱۲۴ رشته‌ها(گره های فرستنده) را ساخته و اجرا می‌کنیم. سپس در خط ۱۲۸ به میزان مشخصی صبر می‌کنیم تا رشته‌ها اجرا شوند. سپس Throughput محاسبه شده در خروجی چاپ می‌شود. همچنین تعداد فریم‌های کل، از دست رفته و موفق نیز چاپ می‌شود. میزان Troughput برای راحتی کاربر بر حسب kbps چاپ می‌شود. در خط ۱۳۷ برنامه اصلی و تمام رشته ها از بین می‌روند.

**قسمت الف**

برنامه را برای ۴۰۰ ثانیه با Tfr = 10 msec اجرا کردیم. نتایج به شرح زیر است:

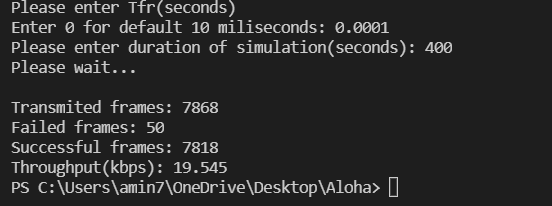
میزان Trhoughput برابر با 18.7975 kbps می‌باشد. همچنین قابل ذکر است با توجه به مقدار Tfr و نرخ ارسال 100kbps هر گره، هر فریم اندازه‌ای برابر با هزار بیت دارد.

**قسمت ب**

**بخش اول**

در صورت سوال گفته شده با فرض ثابت بودن بقیه متغیرها طول زمان ارسال فریم را بگونه‌ای تغییر دهیم که Throughput حداکثر شود. اما این فرض صحیح نیست زیرا این مقدار به اندازه فریم و نرخ ارسال گره ها وابسطه است. یعنی برای تغییر تغییر طول زمان ارسال فریم حداقل یکی از این دو مقدار باید تغییر کند.

اگر طول زمان ارسال فریم برابر با صفر باشد، میزان زمانی که کانال ارسال توسط گرهی اشغال نیز صفر می‌شود. طول زمان ارسال فریم برابر با صفر ایده آل است و در عمل امکان پذیر نیست. بدین جهت Tfr را عددی بسیار کوچک(0.0001) در نظر می‌گیریم و شبیه سازی را برای ۴۰۰ ثانیه اجرا می کنیم. نتایج به شرح زیر است.



بخش دوم

حال فرض می‌کنیم طول زمان ارسال فریم ثابت (۱۰ میلی ثانیه) است و تعداد گره‌ها را تغییر می‌دهیم. در قسمت الف با وجود ۱۰ گره Throughput برابر با 18.7975 kbps اندازه گیری شد. اکنون برای تعداد مختلف گره‌ها برنامه شبیه سازی را اجرا می‌کنیم. نتایج در جدول زیر برای اجرا شبیه سازی به مدت ۴۰۰ ثانیه به شرح زیر است:

|  |  |
| --- | --- |
| Throughput (kbps) | تعداد گره‌ها |
| 9.455 | ۵ |
| 18.7975 | ۱۰ |
| 28.93.75 | ۱۵ |
| 42.1175 | ۲۲ |
| 73.3425 | ۴۰ |
| 81.61 | ۸۰ |
| 93.2975 | ۱۶۰ |
| 70.4075 | ۲۴۰ |
| 75.7275 | ۳۲۰ |

پایان