



دانشکده‌ی مهندسی کامپیوتر

مهر ۱۳۹۴

Computer Networks (شبکه‌های کامپیوتری)

پروژه اول (سیستم اشتراک فایل توزیع شده)

موعد تحویل: ۲۰ آذر

مدرس: مهدی جعفری

هدف: آشنایی با برنامه‌نویسی سوکت با پیاده‌سازی یک برنامه ساده به اشتراک گذاری فایل و تحلیل رفتار و کارایی شبکه با استفاده از آن.

۱ پیاده‌سازی

برنامه شما برای اجرا دو آرگومان را از ورودی دریافت می‌کند. اولین آرگومان اگر "s" باشد، برنامه در نقش سرور و اگر "c" باشد در نقش کلاینت عمل می‌کند. دومین آرگومان پورتهای است که برنامه روی آن به ارتباطات ورودی گوش می‌دهد. برای مثال اگر بخواهید برنامه را در نقش سرور اجرا کنید به نحوی که روی پورت ۵۰۰۱ گوش دهد باید آن را به صورت زیر اجرا کنید:

```
./app s 5001
```

برنامه شما پس از اجرا مانند یک محیط shell کار می‌کند که ارتباطات ورودی را قبول کرده و همزمان یک رابط کاربری را برای ورود دستورات زیر و نمایش نتایج فراهم می‌کند. به منظور سازگاری برنامه‌ها با یکدیگر، فرمت پیام‌های مبادله شده مشخص شده است. این پیام‌ها در موارد لزوم، یکی از انواع REGISTER, CONNECT, DOWNLOAD, UPLOAD, INFO هستند که فرمت بدنه هر کدام در بخش مرتبط آورده شده است.

HELP

لیست دستورات ممکن را برای اجرا روی سیستم فعلی نمایش می‌دهد.

INFO

آدرس آی‌پی برنامه، نقش آن و پورتهای آن گوش می‌دهد را نمایش می‌دهد.

REGISTER ServerIP PortNo

این دستور برای اینکه کلاینت خود را در سرور ثبت‌نام کند و در ضمن آدرس آی‌پی و پورت همه کلاینت‌هایی که در حال حاضر در سرور ثبت‌نام شده‌اند را دریافت کند به کار می‌رود. دو آرگومان ورودی آدرس آی‌پی سرور و شماره پورت آن هستند. هر کلاینت با ارسال یک پیام TCP محتوی شماره پورت خود، در سرور ثبت نام می‌کند. محتوای این پیام به شکل زیر است:

```
REGISTER port#
```

سرور لیستی از آدرس آی‌پی و شماره پورت کلاینت‌ها نگهداری می‌کند و هر زمان کلاینت جدیدی ثبت‌نام می‌کند یا کلاینت از پیش ثبت‌نام شده‌ای به هر دلیل ارتباطش قطع می‌شود، ضمن به‌روزرسانی این لیست، آن را برای همه کلاینت‌های ثبت‌نام شده ارسال می‌کند. محتوای پیام آرسالی به شکل زیر است:

```
IPPORTLIST
IPAddr1 PortNo1
IPAddr2 PortNo2
IPAddr3 PortNo3
...
EOF
```

کلاینت‌ها همواره به رسیدن این به‌روزرسانی‌ها گوش می‌دهند و لیست محلی خود را به‌روز نگه می‌دارند. با دریافت هر به‌روزرسانی توسط کلاینت، این لیست محلی باید نمایش داده شود. توجه کنید که این دستور فقط در کلاینت‌ها کار می‌کند. کلاینت‌های ثبت‌نام شده باید همواره یک اتصال TCP زنده با سرور داشته باشند.

CONNECT PeerIP PortNo

این دستور برای برقراری ارتباط بین دو کلاینت ثبت‌نام شده به کار می‌رود. سمت دیگر ارتباط با آرگومان‌های PeerIP و PortNo مشخص می‌شود. آی‌پی و پورت مشخص شده باید در جدیدترین لیست ارسالی از سمت سرور وجود داشته باشد در غیر این صورت باید از اتصال جلوگیری و پیغام خطای مناسب نمایش داده شود. اتصال به خود و اتصال دوباره نیز غیرمجاز است. هر کلاینت می‌تواند تا حداکثر ۳ اتصال با همتایان خود داشته باشد و باید از اتصالات بیشتر جلوگیری شود. محتوای این پیغام به شکل زیر است که شماره پورت ارسالی، همان پورتهی است که کلاینت فعلی روی آن گوش می‌دهد:

```
CONNECT port#
```

LIST

این دستور لیستی شماره‌گذاری شده از همه ارتباطاتی که نمونه فعلی برنامه جزئی از آن است را نمایش می‌دهد. هر سطر شامل شماره ارتباط، آدرس آی‌پی و شماره پورت همتای مربوطه است. اولین سطر باید ارتباط با سرور و سطرهای دیگر با سایر کلاینت‌ها را نشان دهد. نمونه‌ای از خروجی در اینجا آمده است.

ConnID	IP_Addr	Port_No
1	192.168.69.131	5001
2	192.168.69.132	5003
3	192.168.69.132	5002
4	192.168.69.133	5004

TERMINATE ConnectionID

این دستور ارتباط با شناسه ورودی را قطع می‌کند. هر قطع ارتباط باید با پیغامی در دو سوی ارتباط نمایش داده شود.

EXIT

همه ارتباطات برنامه قطع و رکورد مربوطه از لیست نگهداری شده در سرور حذف و لیست جدید به سایر کلاینت‌ها اطلاع داده می‌شود. کلاینت‌ها با دریافت لیست جدید از سرور، آن را نمایش می‌دهند.

DOWNLOAD ConnectionID FilePath

این دستور فقط در کلاینت‌ها قابل اجراست و فایل موجود در مسیر آرگومان ورودی را از سیستم مشخص شده در ارتباط آرگومان اول دریافت می‌کند و در همان فولدري که برنامه در حال اجراست ذخیره می‌کند. برای مثال اجرای دستور زیر فایل test.txt را در مسیر مشخص از سیستم با آی‌پی 192.168.69.133 می‌گیرد:

```
DOWNLOAD 4 /home/ubuntu/Desktop/test.txt
```

محتوای پیغام ارسالی به شکل زیر است:

```
DOWNLOAD pathToFile
```

هنگامی که دانلود به پایان می‌رسد در دو سوی ارتباط پیغامی متناسب نمایش داده می‌شود. اگر به دلیلی دانلود با مشکل مواجه شود باید پیغام خطا (مثلا فایل مشخص شده وجود ندارد) در دو سوی ارتباط نمایش داده شود.

UPLOAD ConnectionID FilePath

این دستور فقط در کلاینت‌ها قابل اجراست و فایل موجود در مسیر آرگومان ورودی در سیستم فعلی را در سیستم مشخص شده در ارتباط آرگومان اول بارگذاری می‌کند. سیستم سمت مقابل فایل را در همان فولدري که برنامه در حال اجراست ذخیره می‌کند. برای مثال اجرای دستور زیر فایل test.txt را در سیستم با آی‌پی 192.168.69.133 قرار می‌دهد:

```
UPLOAD 4 /home/ubuntu/Desktop/test.txt
```

محتوای پیغام ارسالی به شکل زیر است:

UPLOAD pathToFile fileSize Chunks# ↵

و سپس ارسال chunk های فایل آغاز می شود. هنگامی که آپلود به پایان می رسد در دو سوی ارتباط پیغامی متناسب نمایش داده می شود. اگر به دلیلی آپلود با مشکل مواجه شود باید پیغام خطای متناسب در دو سوی ارتباط نمایش داده شود. توجه: برای خواندن فایل از بافر به حجم ۱۰۰ تا ۱۴۰۰ بایت استفاده کنید و در هر مرحله به اندازه سایز بافر از سوکت بفرستید و ادامه دهید (کل فایل را یکجا نخوانید) در قسمت بعد در این مورد بیشتر توضیح داده می شود.

توجه: در قسمت های قبل برای تبادل پیغام های اطلاع رسانی مانند ثبت نام، اتصال و ارسال فایل (نا) موفق و غیره از پیغام زیر استفاده کنید:

INFO message ↵

۲ تحلیل

برای این قسمت باید برنامه خود را برای مقادیر مختلف سایز فایل در بازه ۱۰۰۰ بایت تا ۱۰ مگابایت و برای سایزهای ۵۰، ۷۵، ۱۰۰ و ۱۵۰ مگابایت تست کنید. همچنین باید سایز بافر را از ۱۰۰ بایت تا ۱۴۰۰ بایت تغییر دهید. برای ایجاد فایل به سایز دلخواه در Unix می توانید از دستور زیر استفاده کنید که یک فایل به سایز ۱ مگابایت می سازد:

```
dd if=/dev/urandom of=testFile.dat bs=1024 count=1024
```

۱.۲ نمودار نرخ ارسال و دریافت داده در برابر سایز فایل

برای این قسمت سایز بافر را ۱۰۰۰ بایت فرض کنید و سایز فایل ارسالی/دریافتی را به مقادیر مختلف گفته شده تغییر دهید و نرخ ارسال/دریافت را محاسبه کنید. مشاهدات خود را گزارش کنید. با تغییر سایز فایل، چه تغییراتی برای نرخ داده انتظار داشتید؟ آیا این تغییرات با اندازه گیری شما هم خوانی دارند؟ اگر خیر، چرا؟

۲.۲ نمودار نرخ ارسال و دریافت داده در برابر سایز بافر

برای این قسمت سایز فایل ارسالی/دریافتی را ۱۵۰ مگابایت فرض کنید و سایز بافر را از ۱۰۰ تا ۱۴۰۰ بایت با مقادیر افزایشی ۲۰۰ بایتی تغییر دهید. مشاهدات خود را گزارش کنید. با تغییر سایز بافر چه تغییراتی برای نرخ داده انتظار داشتید؟ آیا این تغییرات با اندازه گیری شما هم خوانی دارند؟ اگر خیر، چرا؟

۳ نکات مهم

۱. استفاده از کتابخانه های آماده مجاز نیست.
۲. فقط یک برنامه وجود دارد. نسخه های متفاوت برای کلاینت و سرور ارسال نکنید.
۳. کارکرد برنامه در هر شرایطی باید با نمایش پیغام های مناسب قابل فهم باشد.
۴. فایل های تست خود را نفرستید!

موفق باشید