



## پروژه اول درس



دانشکده مهندسی برق و کامپیوتر

سیستم عامل، پاییز ۱۴۰۱

استاد:

مهلت تحویل:

طراحان:

دکتر مهدی کارگهی

یکشنبه ۱ آبان

پارسا کوت‌زری - علی زارع

هدف از انجام این پروژه آشنایی با فراخوانی‌های سیستمی زبان C و یادگیری مبانی socket programming است.

### سوکت چیست؟

سوکت یک مکانیزم برای برقراری ارتباط بین دو پردازنده<sup>۱</sup> روی یک یا چند ماشین است. در این ارتباط دو طرفه، سوکت مثل یک پایانه است که ما اطلاعات را به آن می‌فرستیم یا از آن دریافت می‌کنیم. در واقع سوکت نوعی abstraction برای لایه‌های پایین‌تر سیستم‌عامل است که این ارتباط را ممکن می‌کند.

### شرح پروژه:

در این پروژه می‌خواهیم یک سیستم خرید و فروش نیازمندی‌ها تحت خط فرمان<sup>۲</sup> با استفاده از socket programming و فراخوانی‌های سیستمی زبان C پیاده‌سازی کنیم.

<sup>۱</sup> process

<sup>۲</sup> Command line

## نحوه اجرای برنامه:

در این پروژه هر شخص به عنوان خریدار یا فروشنده وارد سیستم میشود. هر شخصی که به عنوان خریدار وارد میشود ابتدا نام خود را وارد میکند سپس روی یک پورت مشخص که به عنوان آرگومان در خط فرمان دریافت کرده است شروع به گوش دادن میکند تا آگهی‌های جدید و وضعیت آنها را دریافت کند و سپس میتواند روی آگهی‌های موجود درخواست خرید بدهد. دقت کنید اضافه شدن هر فروشنده و یا خریدار نیازمند اجرای یک پردازش<sup>3</sup> جدا است. هر فروشنده‌ای که وارد میشود ابتدا نام خود را وارد میکند سپس عنوان آگهی خود را به همراه یک پورت وارد میکند و آن را از طریق ارتباط UDP روی یک پورت مشخص broadcast میکند تا تمام خریداران این آگهی را دریافت کنند. سپس هر یک از خریداران میتواند روی آگهی‌های موجود که پیشنهاد قیمت بدهد. ارتباط شکل گرفته بین فروشنده و خریدار از نوع TCP صورت میگیرد و خریدار از طریق پورتی که فروشنده به همراه آگهی ارسال کرده به فروشنده که در حال گوش کردن است متصل میشود. بنابراین به طور کلی در فرایند فروش که در آن فروشنده آگهی خود را broadcast میکند خریداران سرور و فروشنده کلاینت محسوب میشود و در زمان خرید که خریداران برای آگهی مورد نظر خود پیشنهاد قیمت میدهند فروشنده به عنوان سرور عمل میکند و خریداران در نقش کلاینت به آن وصل میشوند.

هر آگهی ۳ وضعیت دارد: در انتظار خریدار - در حال مذاکره - منقضی شده

در زمانی که آگهی توسط فروشنده برای خریدارها broadcast میشود وضعیت آگهی در حالت در انتظار خریدار قرار دارد و هرکدام از خریدارها میتوانند با فروشنده ارتباط برقرار کنند و پیشنهاد قیمت بدهند. زمانی یکی از خریدارها برای یک آگهی پیشنهاد قیمت بدهد وضعیت آگهی از حالت در انتظار خریدار به حالت در حال مذاکره تغییر میکند و خریدار دیگری نمیتواند برای این آگهی پیشنهاد قیمت ارسال کند. در صورتی که مذاکره موفقیت آمیز باشد و پیام

---

<sup>3</sup> process

اتمام معامله از طرف فروشنده به خریدار ارسال شود وضعیت آگهی به حالت منقضی شده تغییر میکند و در صورت به نتیجه نرسیدن مذاکره وضعیت آگهی به حالت در انتظار خریدار برمیگردد.

هر خریداری که در حال مذاکره نباشد باید لیستی از عنوان و وضعیت آگهی‌هایی که در انتظار خریدار و در حال مذاکره هستند را ببیند و هر زمان تغییری در وضعیت آگهی اتفاق بیفتد به صورت broadcast به تمام خریدارها اعلام میشود تا لیست خود را آپدیت کنند. در زمان مذاکره بعد از اینکه فروشنده پیشنهاد قیمت را از خریدار دریافت کرد میتواند یکی از ۲ عمل زیر را انجام دهد:

- آن را تایید کند که در این صورت فرایند معامله به اتمام رسیده و آگهی منقضی میشود.
- آن را رد کند که در این صورت فرایند مذاکره به اتمام رسیده و وضعیت آگهی به حالت در انتظار خریدار برمیگردد.

در صورت موفق بودن مذاکره بعد از منقضی شدن هر آگهی فروشنده مربوطه عنوان آگهی را به همراه قیمت نهایی در یک فایل مخصوص به خود لاگ میکند.

فروشنده میتواند همزمان روی چند آگهی با خریداران متفاوت در حال مذاکره باشد بنابراین باید در هر لحظه لیستی از آخرین پیشنهادات خریداران در صفحه فروشنده نمایش داده شود و با رسیدن پیشنهاد جدید لیست آپدیت شود تا تغییر وضعیت پیشنهادات به فروشنده نمایش داده شود و فروشنده با مشخص کردن خریدار و پیامی که میخواهد به او ارسال کند پاسخ هریک از خریداران را بدهد.

### تایمر:

در مرحله مذاکره در صورتی که خریدار تا ۱ دقیقه پاسخی از طرف فروشنده دریافت نکرد مذاکره قطع شده و وضعیت آگهی به در انتظار خریدار باز میگردد.

برای پیاده‌سازی تایمر باید از سیگنال SIGALRM و سیستم کال alarm استفاده کنید.

## همزمانی سیستم:

در کل طول اجرای برنامه، فروشنده میتواند روی چند آگهی با خریداران متفاوت در حال مذاکره باشد بنابراین پردازش فروشنده باید بتواند به طور همزمان به چندین خریدار و درخواستهای آنها رسیدگی کند. ولی برخی از سیستم‌کال‌ها حالت blocking دارند و اجرای برنامه آنجا متوقف می‌شود. برای حل این مشکل از سیستم‌کال select استفاده می‌کنیم. این سیستم‌کال می‌تواند ارتباطات و I/O ها را بدون بلاک کردن مدیریت کند. در این پروژه هم باید به کمک سیستم‌کال select، تمام I/O ها باید بدون اینکه روند اجرای برنامه بلاک شود انجام شوند.

## نکات مهم:

- فرمت و محتوای نمایش داده شده برای هریک از انواع پیام‌ها به عهده خودتان است و هر نوع نمایش معقولی که محتوای پیام‌های رد و بدل شده به درستی نمایش دهد قابل قبول است.
- میتوانید فرض کنید اسامی استفاده شده برای اشخاص و آگهی‌ها یکتا هستند و هیچ فروشنده‌ای آگهی با عنوان یکسان را دوبار سابمیت نمی‌کند بنابراین نیازی به چک کردن این موارد نیست.
- فروشنده تنها زمانی میتواند آگهی جدید ایجاد کند که روی هیچ آگهی با وضعیت در حال مذاکره نداشته باشد.
- برای راحتی در پیاده‌سازی میتوانید فرض کنید زمانی که مذاکره خریدار به اتمام میرسد وضعیت آگهی‌هایی که قبلاً دریافت کرده است پاک میشود و خریدار تنها به آگهی‌هایی که بعد از اتمام مذاکره اضافه شده‌اند و یا تغییر وضعیت داده‌اند دسترسی دارد.
- یک فروشنده میتواند همزمان روی چند آگهی متفاوت در حال مذاکره باشد ولی خریدار میتواند همزمان تنها روی یک آگهی در حال مذاکره باشد. دقت کنید فروشنده نمیتواند همزمان با دو خریدار روی یک آگهی مذاکره کند.

- توجه کنید تغییر وضعیت آگهی باید از طریق ارتباط UDP به تمامی خریداران اطلاع داده شود.
- تمامی آدرس‌های IP را localhost یا همان 127.0.0.1 در نظر بگیرید.
- با قرار دادن stdin در لیستی که به select می‌دهید می‌توانید بدون بلاک شدن از کنسول ورودی بخوانید.
- فروشنده و خریدار باید به این شکل اجرا شوند:

`./buyer <server_port>`

`./seller <server_port>`

## نکات پایانی:

- در این پروژه کدهایتان باید به زبان C باشد و با gcc قابل کامپایل شدن باشد.
- توجه کنید که پروژه‌های درس تک نفره‌اند.
- در حین اجرای برنامه logهای مناسبی مانند وصل شدن خریداران و پیام‌های رد و بدل شده در مذاکرات روی آگهی‌های مختلف برای خریدارها چاپ کنید تا روند اجرای برنامه مشخص باشد. این logها هنگام تحویل بخشی از عملکرد کد را نشان می‌دهند.
- پیاده‌سازی شما باید به کمک سیستم‌کال‌ها مانند read, write, open و ... باشد و استفاده از توابع کتابخانه‌ای مانند fopen مجاز نیست. توابعی که سیستم‌کال هستند در <https://linux.die.net/man/2> قابل مشاهده‌اند.
- توابع کتابخانه‌ای که توسط سیستم‌کال‌ها قابل پیاده‌سازی نیستند مانند atoi, strcat, strcpy, sprintf و ... مجاز هستند.
- برای آشنایی با برنامه‌نویسی سوکت می‌توانید از منابع زیر و ویدیوهایی که در سایت درس قرار داده شده استفاده کنید.

<https://beej.us/guide/bgnet/html/#client-server-background>

<https://beej.us/guide/bgnet/html/#system-calls-or-bust>

<http://beej.us/guide/bgnet/html/#broadcast-packetshello-world>

- فایل نهایی که تحویل می‌دهید باید شامل کدهای توسعه داده شده (ترجیحا به همراه Makefile) باشد که در نهایت بعد از کامپایل کردن دو فایل اجرایی buyer و seller را تولید کنند.  
این فایل ها در قالب یک فایل فشرده zip با نام OS\_CA1\_<SID>.zip در صفحه درس آپلود کنید.
- در صورتی که سوالی داشتید می‌توانید از طریق فروم درس در ایلرن و یا ایمیل به دستیاران آموزشی پروژه سوال خود را بپرسید.

[pkootzari@gmail.com](mailto:pkootzari@gmail.com)

[alizare1@ut.ac.ir](mailto:alizare1@ut.ac.ir)