



سیستم‌های عامل

گزارش تمرین صفر
دانشکده مهندسی کامپیوتر
دانشگاه صنعتی شریف
نیم سال اول ۹۹-۰۰

اساتید:
جناب آقای دکتر خرازی
نام و نام خانوادگی:
امیرمهدی نامجو - ۹۷۱۰۷۲۱۲



۱ راه اندازی اولیه

۱.۱ نصب ماشین مجازی

ابتدا Virtual Box و Vagrant را دانلود و نصب کردم. همچنین از آن جایی که روی ویندوز بودم و از قبل Cygwin را نصب داشتم، کنترل کردم که کتابخانه های gcc و مواردی نظیر این که بعدا ممکن است لازم باشد نصب باشند.

برای اجرای vagrant به مشکلی خوردم که در حالت vagrant up در مرحله ssh متوقف می شد. بعد از جست و جو در اینترنت به این صفحه در [StackOverflow](https://stackoverflow.com) رسیدم. البته یکی از دانشجویان درس هم یک فایل Vagrantfile در دیسکورد قرار داده بود ولی همچنان با آن هم مشکل داشتم. طبق راه حل این صفحه، چند خط دیگر هم باید اضافه می کردم و در نهایت این خطوط به Vagrantfile اضافه شدند.

```
config.vm.provider "virtualbox" do |vb|
  vb.gui = true
  vb.customize ["modifyvm", :id, "--uart1", "0x3F8", "4"]
  vb.customize ["modifyvm", :id, "--uartmode1", "file", File::NULL]
  vb.customize ["modifyvm", :id, "--nestedpaging", "off"]
  vb.customize ["modifyvm", :id, "--cpus", 4]
  vb.customize ["modifyvm", :id, "--paravirtprovider", "hyperv"]
  vb.customize [ "modifyvm", :id, "--cableconnected1", "on" ]
end
```

که البته قسمت gui تاثیری ندارد و صرفا باعث می شود لاگ های سیستم عامل در یک پنجره مجزا دیده شوند. حضور بقیه موارد همگی لازم بود تا در نهایت vagrant up به درستی کار کند. بعد از آن vagrant ssh هم بدون مشکل اعمال شد و به ماشین مجازی متصل شدم و دستورات مربوط به نصب پیچ را اجرا کردم.

۲.۱ Git

ابتدا به سامانه ترشت لاگین کردم و محیط آن را بررسی کردم. برای قسمت Git در ماشین مجازی، دو دستور گفته شده را به صورت زیر اجرا کردم.

```
git config --global user.name "Amirmahdi Namjoo"
git config --global user.email "amirm137878@gmail.com"
```

پس از آن با دستورات گفته شده در داک کليدهای عمومی و خصوصی را تولید کرده و در سامانه ترشت قرار دادم. بعد از آن قسمت handouts و همچنین بخش فردی را از گیت دریافت کردم.



۳.۱ اتصال VSCode به ماشین مجازی

پس از آن از آن جایی که می خواستیم از VSCode به عنوان ادیتور گرافیکی استفاده کنیم، با جست و جویی مختصر راه حل اتصال ssh آن به vagrant را پیدا کردم. برای این کار، باید در پوشه ای که Vagrantfile هست، دستور vagrant ssh-config را اجرا کنیم. سپس متن تولید شده را باید در تنظیمات vscode قرار داد. برای این کار در command palette نرم افزار ssh را تایپ می کنیم و فایل configuration آن را باز کرده و متن نوشته شده را که (در ویندوز) شبیه متن زیر است، کپی می کنیم.

```
Host vagrant
HostName 127.0.0.1
User vagrant
Port 2222
UserKnownHostsFile /dev/null
StrictHostKeyChecking no
PasswordAuthentication no
IdentityFile C:/Users/username/.vagrant.d/boxes/ce424-VAGRANTSLASH-spring2020/1.0.0/virtualbox/vagrant_private_key
IdentitiesOnly yes
LogLevel FATAL
ForwardAgent yes
ForwardX11 yes
```

سپس با تایپ مجدد دستور ssh و انتخاب گزینه وصل شدن به Remote-ssh و انتخاب vagrant، به vagrant وصل شدم. (این که چه نامی ظاهر بشود، بستگی به عبارت ابتدای متن بالا یعنی Host vagrant دارد. این نام را می توان یک نام کاملاً دلخواه در نظر گرفت.)

۲ موارد مربوط به یادگیری

پس از این موارد، کمی در مورد gdb، make و کاربردهای پیچیده تر git که البته فعلاً در این تمرین نیازی نشده است مطالعه کردم. از vim در حد خیلی کوتاه استفاده کردم ولی از آن جایی که خیلی با آن راحت نیستم، سراغ VSCode رفتم. بقیه موارد را هم در حدی که لازم باشد تست کردم و از man هم در قسمت های مختلف برای مشاهده راهنمای دستورات استفاده کردم. پس از مطالعه موارد اصلی به سراغ مابقی تمرین رفتم.

۳ تمرین های مقدماتی

۱.۳ کد Words

در این بخش کدهای خواسته شده را پیاده سازی کردم که در گیت قابل مشاهده است. تنها نکته قابل توجه این بود که به نظر فایل آماده wc_sort به شکل عجیبی یک عنصر null را هم در لیست های



سورت شده وارد می کند. خود این موضوع مشکل ساز نیست ولی در چاپ مقادیر frequency به اشتباه یک مقدار صفر null هم چاپ می شد که برای رفع این مشکل، در تابع از پیش پیاده سازی شده چاپ کننده (fprintf_words) تغییر کوچکی ایجاد کردم که اگر مقدار تکرار یک عبارت صفر بود، آن را چاپ نکند. (عملاً معنی هم ندارد چیزی که صفر بار تکرار شده چاپ شود)

همچنین در هنگام نوشتن این داک به یک باگ حیاتی برخوردیم که کدم به دلیل وجود دستور rewind در هنگام خواندن از روی stdin، به مشکل می خورد که آن را برطرف کردم و بدون دستور rewind پیاده سازی کردم. در اصل اگر عملگر > فایل به عنوان stdin داده می شد یا این که کلا یکسری آرگومان فایل داده می شد هیچ مشکلی پیش نمی آمد، اما اگر داده ها را دستی وارد می کردم، نیاز به دو بار وارد کردن داده ها و دو بار EOF زدن بود که مشکل را حل کردم. همچنین با جست و جو فهمیدم که در هنگام ورودی دادن به stdin در سیستم های UNIX-Based با استفاده از Ctrl+D و در ویندوز با Ctrl+Z می توان کاراکتر خاص EOF را ارسال کرد.

۱.۱.۳ make

در مورد توضیحات make، این موارد در Makefile.txt قرار گرفته اند. موارد مختلفی که لازم بود را با جست و جو در راهنماهای آماده و همچنین داکيومنت های قرار داده شده در قسمت یادگیری پیدا کردم.

۲.۳ limits

برای این قسمت، با استفاده از man به جزییات getrlimit پی بردم و فهمیدم که برای حل این سوال، باید به عنوان پارامتر اول ورودی یکسری Constant تعریف شده خاص را پاس بدهم که برای این سوال، موارد RLIMIT_STACK و RLIMIT_NPROC و RLIMIT_NOFILE بودند. همچنین ابتدا مقدار rlim_max را داشتیم می گرفتیم که مشاهده کردم برای استک، مقدار $2^{64} - 1$ چاپ می شود. سپس با داک دانشگاه برکلی (!) مراجعه کردم و متوجه شدم که ظاهراً این سوال soft limit را می خواهد و نه hard limit. این موضوع در متن فارسی به نظر واضح ذکر نشده بود که کدام مقدار حدی را می خواهیم. به هر حال با توجه به این که فهمیدم soft limit را می خواهیم از rlim_cur استفاده کردم. همچنین makefile نوشته شده برای سوال برای target های اصلی یعنی map و limit به خود map.c و limit.c وابسته نشده بود و متوجه تغییرات آن ها نمی شد و صرفاً بررسی می کرد که فایل های باینری تولید شده باشند که با توجه به این موضوع، تغییرات جدید سورس کد را کامپایل نمی کرد. البته من هم آن را دستکاری نکردم و برای بازسازی فایل ها، ابتدا فایل هایی که از قبل کامپایل شده بودند را حذف می کردم که متوجه بشود دستور را دوباره اجرا کند.

۳.۳ gdb

برای این قسمت دوباره در مورد gdb مطالعه کردم و سپس آن را حل کردم. جزییات کامل در gdb.txt موجود است.

۴.۳ Compiling, Assembling, and Linking

برای این قسمت ها کارهای گفته شده را انجام داده و نتیجه را در call.txt نوشته ام. تنها نکته این بود که با مراجعه به داک دانشگاه برکلی متوجه شدم قسمت دوم سوال به اشتباه نوشته شده در مورد section. توضیح بدهیم که اصلاً چنین چیزی وجود ندارد و هر کدام از بخش هایی نظیر .data،



rodata، .text و ... یک section هستند. با پرسیدن سوال در دیسکورد و گرفتن تاییدیه وجود مشکل، توضیحات را در مورد data و .text نوشتم.

۵.۳ نکته پایانی در مورد این داک

این \LaTeX روی ماشین مجازی کامپایل نشده است. با این حال برای کامل بودن تمرین، فایل‌های سورس آن را به vm انتقال دادم و روی گیت هم ارسال کردم.