

## نصب پیش‌نیازها:

طبق توضیحات داک `vagrant` , `virtual box` را بر روی اوبونتو نصب کردم و تنها در اجرای دستور `vagrant up` به ارور 404 برمی‌خوردم که به خاطر وجود تحریم‌ها بود. به صورت دستی تصویر ماشین مجازی را از `vagrant cloud` دانلود کردم و با استفاده از دستور `vagrant box add` آن را اضافه کردم و سپس دستور `vagrant up` را اجرا کردم و همه چیز درست پیش رفت.

سپس `ssh key` ماشین لوکال را به `authorized_keys` ماشین مجازی اد کردم و سپس با استفاده از افزونه‌های `vscode` به ماشین مجازی دسترسی پیدا کردم.

## :git

`ssh key` ماشین مجازی را با سامانه طرشت اضافه کردم و کانفیگ‌های `git` را تنظیم کردم. با توجه به اینکه دسترسی به سامانه طرشت دیر داده شد اندکی در ست آپ کردن ریپوها مشکل ایجاد شد فولدر `git` را که کامیت‌ها را به مرور می‌زد پاک شد و مجبور شدم روند اجرای تمرین را دوباره از اول کامیت کنم.

## :words

ابتدا مفاهیم پوینتر و استراکت را از داکيومنت‌های مختلف مرور کردم و سپس تابع‌های پرونده‌ی `word_count.c` را کامل کردم. چالشی در این بخش نداشتم اما در بخش کامل کردن تابع‌های پرونده‌ی `main.c` ابتدا کدهای از پیش آماده شده را خواندم و بعضی مفاهیم را با آنها آشنا نبودم و درمورد آنها سرچ کردم. در ادامه نیز تابع‌های مورد نیاز در این پرونده را تکمیل کردم. ابهامی که در این سوال وجود داشت این بود که به صورت دقیق در سوال ذکر نشده بود که مفهوم کلمه چه چیزی است. بدین معنی که عبارت `a12c` یا 123 نیز کلمه حساب می‌شوند یا نه. من با استناد به خط اول صفحه ۱۰ داک سوالات که از عبارت "شناسه‌های الفبایی" استفاده کرده است تنها مجموعه کاراکترهایی که شامل تنها حروف الفبایی هستند را به عنوان `word` حساب کردم.

## :make

با توجه به آشنایی قبلی ای که با `Makefile` ها داشتم خیلی چالشی در این تمرین نداشتم و صرفاً کمی از داکيومنت‌های درون داک تمرین را خواندم و توضیحات تکمیلی برای این تمرین در فایل `makefile.txt` درون فولدر آمده است.

## :Userlimit

در این سوال ابتدا `man getrlimit` را مطالعه کردم و سپس بعد از آشنا شدن با توابع مختلف آن و انواع ریسورس تایپ‌ها در آن توابعی که خواسته‌های مسئله را برآورده می‌کنند را استفاده کردم. نتایج را چاپ کردم. توجه کنید که استراکت `rlimt` دو اتریبیوت دارد که یکی از آنها `softlimit` و دیگری `hardlimit` است. با توجه به خروجی نمونه در صورت سوال و همچنین چاپ کردن مقدار اتریبیوت‌های دیگر متوجه شدم که خواسته‌ی سوال در این سوال همان `softlimit` است که در اتریبیوت `rlim_cur` ذخیره می‌شود.

## :gdb

در این سوال ابتدا [این داک](#) در مورد gdb را به طور خلاصه در مورد gdb مطالعه کردم و سپس مرحله به مرحله سوالات را انجام دادم. یکی از مشکلاتی که داشتم این بود که هنگام اجرای دستور layout، صفحه‌ی ترمینال کرش می‌کرد که مجبور شدم در لوکال این بخش را انجام دهم و از دستورهایی برای مشاهده اسمبلی کد نیز استفاده کردم.

## **:Compiling, assembling, and linking**

در بخش دو این سوال یک اشتباه وجود دارد که بخشی به اسم section وجود ندارد و تی ای مربوطه نیز در دیسکورد به این نکته اشاره کرد و آن را اصلاح کرد.

در این سوال ابتدا خواند man objdump بسیار کمک‌کننده بود. به خصوص در پیدا کردن فلگ‌هایی که باید در هر قسمت با دستور پاس داده می‌شد تا عمل مورد نیاز را انجام دهد. به جز خواندن کدهای اسمبلی چالش دیگری نداشتم که مجور شدم اندکی اسمبلی و سینتکس ها را مرور کنم.

## **:report**

برای تحویل دادن این فایل ریپورت در گیت مجبور شدم که این pdf را به داخل ماشین مجازی انتقال دهم تا آن را کامیت و پوش کنم.

برای این کار از دستور scp report.pdf vagrant:/home/vagrant/code/personal/HW0 استفاده کنم.(vagrant) (host define in .ssh/config)