فاز اول پروژه سیستم عامل

در این فاز موردی که از ما خواسته شده بود این بودکه فراخوان سیستمی ایجاد کنیم

۱- در فایل syscall.h برای سیستم کال proc_dump عدد ۲۲ را به لیست اضافه میکنیم SYS_proc_dump

۲- در گام دوم می بایست در فایل syscall.c تابع ها مورد نظر مان را در سیستم کال اضافه میکنیم و عدد مورد نظر را نیز همینطور با extern در این فایل وارد میکنیم

۳-در این قسمت نیز ۲ تابع داخل proc.h مینویسیم proc_info که در داکیومنت گفته شده بود و دیگر producttable که از فایل proc.c جدایش کردیم

۴- در اینجا نیاز به این داریم که تابعی بنویسیم که proccesstable را بگیرد به دلیل اینکه در فایل ها دیگر نیاز به این تابع داریم این تابع را نیز باید در defs.h نیز تعریف کنیم و هرجا که به این تابع نیاز داشتیم از آن استفاده کنیم

۵- در مرحله آخر با توجه به proccesstable که گرفتیم به سیستم کال خود که همان proc_dump می باشد دو ورودی می دهیم یکی لیست proc_info و سایز آن که پوینتر هست و این سیستم کال این مقادیر را برای ما پر می کنند. حال با توجه به table که داریم lock میکنیم و بعد از آن process های RUNNING و RUNNABLE را پیدا میکنیم و سورت میکنیم و بعد از این سورت کردن اطلاعاتی که در proc_info ذخیره میکنیم سایز و proccessid و proccess های سورت شده است و در نهایت این proccess هایی که lock را free میکنیم

۶- برای تست این برنامه یک فایل c دیگر به نام test مسازیم و در ۲جا در makefile قرار میدهیم در داخل UPROGS به صورت \test_ و در EXTRA نیز فیل با پسوند c و کاری که در فایل تست انجام می دهیم به این صورت هست که تعدادی proccess را ابتدا فورک میکنیم و با استفاده از malloc فضا های مختلفی ایجاد میکنیم و اشغال میکنیم در حافظه و در یک حلقه بینهایت برای اینکه proccess هامون از حالت RUNNING خارج نشوند قرار میدهیم و یک sleep کوتاه میذاریم که child ها ایجاد شوند و بعد با توجه به ورودی که برای proc_dump در نظر گرفتیم می بایست یک لیست به اندازه حداکثر پردازنده ها می سازیم و به proc_dump پاس میدیم و بعد از چاپ کردن child هارو kill میکنیم