گزارشکار آزمایشگاه یک آزمایشگاه سیستم عامل

سوالات گزارش کار:

1-

مدیریت منابع/مدیریت پردازه ها/واسط بین سخت افزار و نرم افزار

2-

Basic header:دارای constants و convention هایی که xv6 نیاز دارد

Entering xv6:کدی که xv6 را شروع می کند.xv6 برای لانچ شدن نیاز به یک سری ضروریات دارد که در فایل entry است

Locks:هماهنگ سازی دسترسی به ریسورس های مشترک

Processes:چیز هایی که برای انجام پراسس های یوزر و مولتی پروگرمینگ و مولتی تسکینگ نیاز داریم

System calls:کد های مربوط به سیستم کال ها که اینترفیس بین یوزر و کرنل است

File system:استراکچرهایی که برای مدیریت فایل ها دایرکتوری ها و مموری دیسک نیاز داریم در این قسمت است

Pipes:برای ارتباط بین پردازه ها استفاده می شود.پایپ ها به این صورت هستند که یک پراسس در یک طرف آن دیتا را می نویسد و پراسس دیگر در طرف دیگر آن

String operation:شامل عملیات هایی مانند strlen برای استرینگ ها

User level:کد init را دارد که همواره درحال اجرا می باشد.هدر user.h نیز دارای functionality های مربوط به یوزر است

Boot loader:کد هایی به زبان سی و اسمبلی که وظیفه ی آن ها لود کردن کرنل از دیسک به مموری است

Link:

3-

4-

UPROGS مخفف user program است که در آن قسمت می توانیم پروگرم های مربوط به یوزر را ببینیم و پروگرم های خودمان را اضافه کنیم.

ULIB مخفف یوزر لایبرری است.شامل utiliy function ها و سیستم کال هایی است که اجازه می دهند یوزر پروگرم ها با کرنل ارتباط داشته باشند

5-

8- از این دستور برای convert کردن object ها استفاده می شود.در xv6 بوت لودر برای لود کردن مستقیم فایل executable در مموری باید فرمت خاصی داشته باشد که این command آبجکت فایل را می گیرد و به فایل باینری مطلوب تبدیل می کند.

13- با توجه به اینکه بخشی از حافظه ی قبل این آدرس مربوط به bios است باید کرنل در بخشی قرار بگیرد که با bios تداخلی نداشته باشد

18-دلیل استفاده از seguser آن است که بین پردازه های سطح کاربر و پردازه های سطح هسته جداسازی انجام شود.

19-proc struct شامل attribute های زیر است:

Pid,state,sz,pgdir,context,chan,kstack,tf,pid,parent

Pid:

Int

یک شناسه ی یونیک برای هر پردازه

State:

Enum proc\_state

استیت فعلی پردازه را می گوید.برای اسکجلر مهم است که بداند آیا پردازه در حال ران شدن است یا منتظر است یا اصلا متوقف شده

Sz:

Int

سایز پردازه به بایت در مموری که شامل استک و هیپ اختصاص داده شده نیز هست

Pgdir:

Pde\_t\*

پوینتری که به دایرکتوری پیج یک پردازه اشاره دارد.این استراکچر کمک می کند پردازه آدرس های مجازی را به مموری فیزیکی مپ کند

Context:

Struct context

در هنگام context switch باید مقادیر بعضی از رجیستر ها و استیت پردازه ها سیو شود که وقتی دوباره به پردازه برگشتیم بدانیم باید از کجا شروع کنیم.این مقادیر در این متغیر است

Chan:

Void\*

وقتی پردازه ای به حالت sleep می رود کاربردی است که دلیل آن را بدانیم.مثلا دلیلش انتظار برای I/O بوده یا نه.این متغیر شامل این دلیل است.

Kstack:

Char[]

استک هسته برای پراسس که برای کال کردن توابع مورد نیاز است

Tf:

Struct trapframe

Trap frame هر پردازه را دارد که شامل رجیستر ها و بقیه اطلاعات مربوط به استیت پردازه است که برای هندل کردن اینتراپت ها و اکسپشن ها لازم است.

Parent:

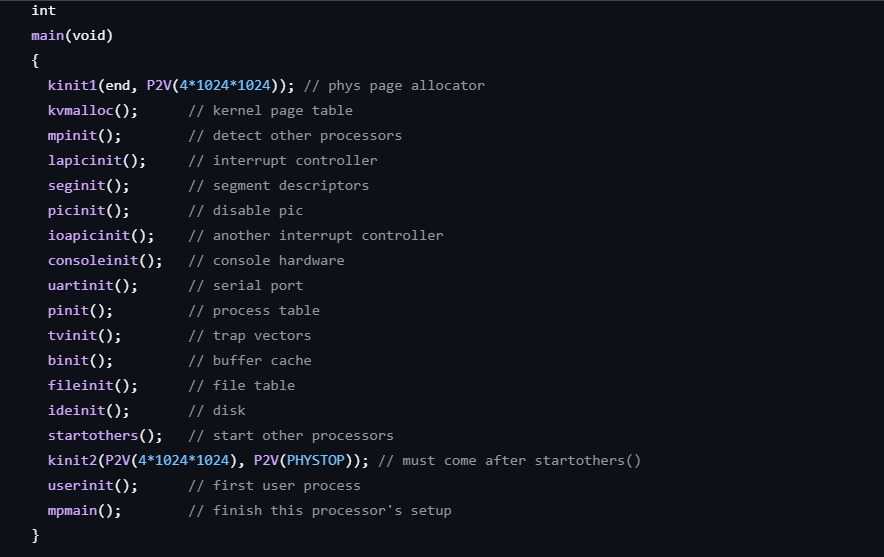
Struct proc\*

پوینتر به پردازه ی والد که برای سلسله مراتب پردازه ها لازم است.

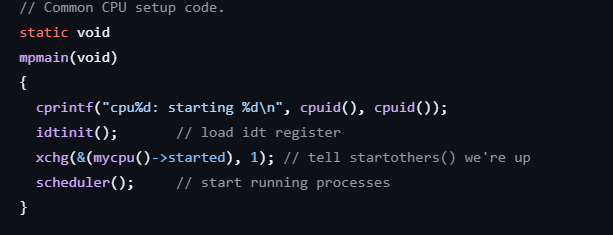
معادل این ساختار در لینوکس task struct است

این استراکچر به طور کلی در مدیریت پردازه ها نقش مهمی دارد.

23-



همانطورکه از بخشی از کد main.c مشخص است تا تابع startothers مختص یک هسته هست و بعد از این تابع بقیه هسته ها شروع به کار می کنند.بنابراین توابعی که در main هستند مختص هسته ی اول است.



همانطور که از کد مشخص است تابع mpmain که به تابع startothers می گوید آماده ی اجرای پردازه هست بین چند هسته مشترک است.

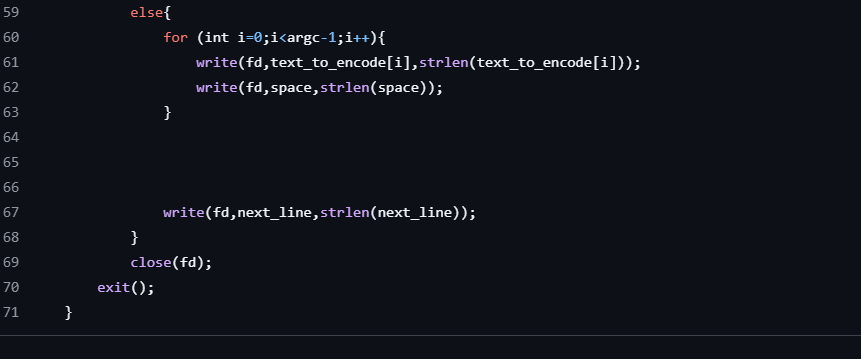
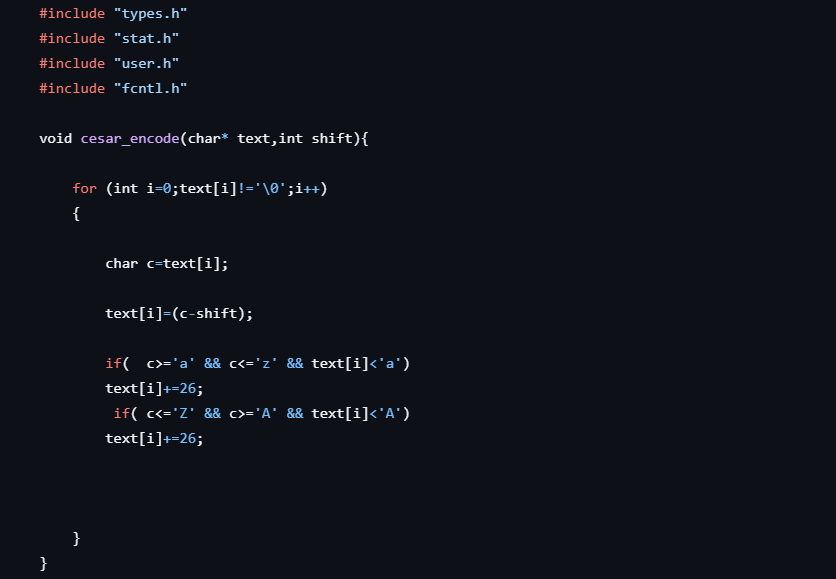
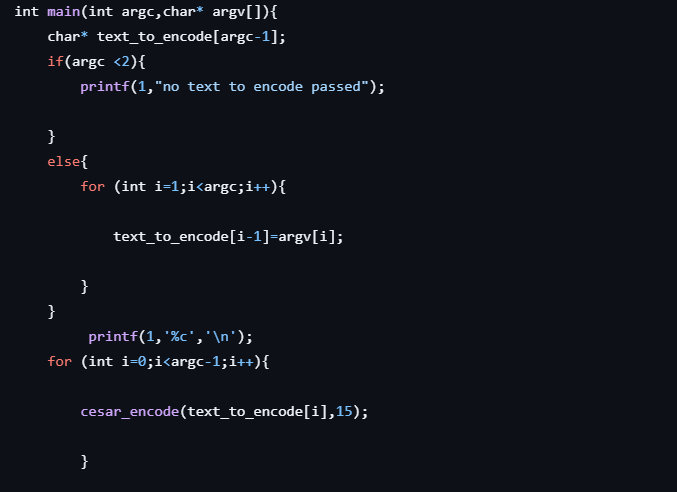
بنابراین:

مرحله ی boot به طور اختصاصی توسط هسته ی اول انجام می شود.بنابراین توابعی مانند kvmalloc,mpinit,startothers,picinit یعنی توابعی که برای آماده سازی سیستم هستند اختصاصا توسط این هسته انجام می شوند.

توابعی مانند mpmain,switchkvm مشترکا توسط همه ی هسته ها اجرا می شوند.

زمانبند یا scheduler در تابع mpmain است پس مشترکا توسط همه ی هسته ها اجرا می شود.

اضافه کردن برنامه ی سطح کاربر:



GDB:

1-

برای مشاهده breakpoints ها از info breakpoints ا استفاده می شود.

2-برای حذف breakpoint ها می توانیم از info breakpoints شماره هر break point را یافته و از دستور del برای حذف آن ها استفاده کنیم.

اگر del را با شماره بزنیم فقط breakpoint با شماره مربوطه حذف می شود و اگر خالی بزنیم همه ی breakpoint ها حذف می شود.

3-دستور bt مخفف backtrace است.این دستور بابت هر فریم استک یک لاین پرینت می کند.

4-