تمرینات سری دوم درس اصول سیستم ای عامل

استاد محمود فصلعلی

سروش حیدری - 96222031

سوال 1----------

برنامه مجموعه ای از دستور العمل هاست که بصورت کد درامده تا هدف خاصی را انجام دهد

پردازه یک برنامه در حال اجرا است

و نخ واحد های اجرایی یک پردازه هستند

بطور کلی یک برنامه در حال اجرا میتواند از چند پردازه در یک زمان تشکیل شده باشند و هر کدام از آن پردازه ها میتوانند از چند نخ بطور جداگانه تشکیل شده باشند

سوال 2----------

زمانی که برنامه ما روی یک سیستم چند پردازشی اجرا شود با داشتن یک پردازه چند نخی برنامه ما میتواند بطور سخت افزاری از چند شمارنده برنامه دستور بگیرد بطوری که هر نخ شماره برنامه خود را داشته و بطور موازی اجرا میشوند (البته تعداد شمارنده برنامه ها محدود به تعداد پردازشگر هاست)

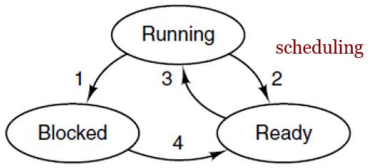
سوال 3----------

زمانی که یک برنامه را بطور بازگشتی اجرا میکنیم برای مثال در تابع فیبوناچی که به شکل بازگشتی نوشه شده برای محاسبه F(x) یک فرایند فرزند تولید میشود تا F(x – 1) را محاسبه کند (و ی فرایند دیگر برای F(x – 2)) اینجا مفهوم Context Switching اتفاق می افتد یعنی وضعیت فعلی فرایند والد که F(x) بود جایی ذخیره میشود و فرایند والدش اجرا میشود وقتی فرایند والد تمام شد دوباره مجبور هستیم زمانی را صرف کنیم تا وضعیت فعلی فرایند والد را بازیابی کنیم

این ذخیره سازی و بازیابیی که در Context Switching بین والد ها و فرزند هایشان رخ میدهد (در یک تابع بازگشتی این اتفاق بسیار رایج است زیرا که هر والد برای خود والدی میسازد و ...) زمانبر است و در استفاده از یک پشته دیگر نیازی به Context Switching به این شکل و این هزینه نداریم و سرعت اجرا بالا تر است (زیرا در زمان Context Switching پردازنده هیچ کار مفید دیگری انجام نمیدهد) اما نکته دیگری که وجود دارد این است که ممکن است استفاده از توابع بازگشتی سهولت بیشتر یا نکات مثبت دیگری نسبت به یک پشته داشته باشد که اینجا مورد بحث ما نیست

سوال 4----------

با توجه به شکل زیر داریم :



در زمان تولید شدن فرایند فرزند فرایند والد در حالت "در حال اجرا" بوده است که با تولید فرایند فرزند دو حالت رخ میدهد

1- برای مثال فرایند فرزند را بجود اورده ایم تا یک ورودی از کاربر گرفته یا یک خروجی را نمایش دهیم, در این حالت فرایند والد وارد حالت "بسته شده" میشود در این حالت منتظر فرایند فرزند شده و منظر میشود ورودی یا خروجی مورد نظرش تکمیل شود سپس وارد حالت "انتظار شده" و منتظر میماند تا توسط CPU Scheduling دوباره پردازنده به آن اختصاص داده شود تا به ادامه ی اجرا بپردازد

2- حالت دیگری که رخ میدهد این است که فرایند فرزند ربطی به دستگاه های ورودی و خروجی ندارد و فرایند والد مستقیما از حالت "در حال اجرا" به حالت "آماده" برود و صرفا منتظر بماند تا فرایند های با اولویت های بالاتر و فرایند های فرزندش به اتمام رسیده تا دوباره به پردازنده دسترسی پیدا کرده و کار خود را ادامه دهد

سوال 5----------

روش سمافور (Semaphores) که گویا بیشتر از روش پیترسون استفاده میشود و طرز کار این روش بر اساس دو عمل اتمیک است (Wait & Signal) که در زیر توضیح داده شده اند

```
wait(S)
{
   while (S<=0);

   S--;
}
```

عمل ()Wait ارگومان S را در صورت مثبت بودن کم میکند تا زمانی که S منفی یا صفر شود

```
signal(S)
{
   S++;
}
```

عمل ()Signal ارگومان S را اضافه میکند

از آنجایی که این دو عمل اتمیک هستند در حین انجام آن ها هیچ فرایند دیگری از پردازشگر استفاده نخواهد کرد و حالت بحرانی در این ببین بوجود نمیآید

زمانی که یک فرایند در حال استفاده از منابع است باید با فراخوانی متغیر بالا بقیه فرایند هارا در حال انتظار بگزارد

توضیحات بیشتر در لینک زیر :
http://www2.cs.uregina.ca/~hamilton/courses/330/notes/synchro/node3.html