

«به نام ایزد یکتا»



## گزارش هشتم درس آزمایشگاه سیستم عامل

استاد: مهندس قاسمی

تهیه کننده: بردیا اردکانیان

۹۸۳۱۰۷۲

### بخش اول) FCFS

در این بخش با توجه به ورود پردازش ها، هر کدام زودتر آمده باشد (طبق ترتیب وارد کردن آن ها در ورودی) اجرا خواهند شد. در پیاده سازی، از این نکته استفاده می کنیم که زمان انتظار هر پردازش برابر است با زمان اجرای فرآیند قبل آن و زمان اجرای آن هم برابر است با مجموع **Burst Time** و زمان انتظار برابر می باشد.

### بخش دوم) SJF

برای این بخش پس از گرفتن ورودی های لازم مانند بخش قبل، پردازش ها را به ترتیب صعودی `cpu burst time` سورت می کنم و سپس آن ها را اجرا می کنم. (این قسمت مانند قسمت قبلی اجرا می شود چون ترتیب پردازش ها را می دانیم).

### بخش سوم) Priority

برای این بخش پس از گرفتن ورودی های الزم مانند ۲ بخش قبل، پردازش ها را به ترتیب صعودی اولویت ها سورت می کنم زیرا هر چه عدد اولویت کمتر باشد، اولویت آن بیشتر است و سپس آن ها را اجرا می کنم.(این قسمت مانند قسمت اول اجرا می شود چون ترتیب پردازش ها را می دانیم).

### بخش چهارم) Round Robin

در این بخش با گرفتن یک quantum time از پردازش مشخص می کنیم که هر پردازش تنها مدت زمان محدودی برای اجرا دارد و سپس نوبت پردازش بعدی می شود و این مدت زمان محدود همان quantum time است.

بخش پنجم) مقایسه

الگوریتم	کاربرد
FCFS	این روش پیاده سازی و منطق آسانی دارد ولی از طرفی برای تعداد پردازش های زیاد باعث قحطی می شود زیرا همواره پردازش های جدید را اجرا می کند و باعث می شود بعضی پردازش ها خیلی منتظر بمانند. این روش برای batch system مناسب است و برای سیستم هایی مثل صف خرید بلیط سینما کاربرد دارد.
SJF	این روش باعث می شود throughput بالا رود زیرا پردازش های کوتاه تر را ابتدا اجرا می کند و معمولاً در سیستم تعداد پردازش های کوتاه بیشتر از طولانی است. در اینجا ولی نیز قحطی داریم و باعث می شود پردازش های طولانی هیچگاه اجرا نشوند. این روش باعث می شود Average Waiting Time کاهش یابد همانطور که در این آزمایش دیدیم که کمترین Average Waiting Time را داشت پس برای Long term scheduling مناسب است در پردازش ها با تعداد بالا. این روش برای batch system مناسب است.
Priority	این روش برای زمانی مناسب است که برای هر پردازش اولویت آن را بدانیم و در آن صورت باعث می شود پردازش ها با اهمیت بیشتر برای ما ابتدا اجرا شوند، البته در اینجا نیز احتمال قحطی وجود دارد ولی می توانیم با الگوریتم aging این موضوع را کنترل کنیم. این روش برای Interactive system مناسب است
Round Robin	این روش باعث می شود که turnaround ما به صورت میانگین افزایش یابد ولی به صورت کلی response time آن کمتر است. البته باید به این موضوع توجه کرد که اگر q بزرگ باشد شبیه به همان fcfs می شود و اگر کوچک باشد، overhead زیادی خواهیم داشت. این روش برای interactive system مناسب است