Operating Systems

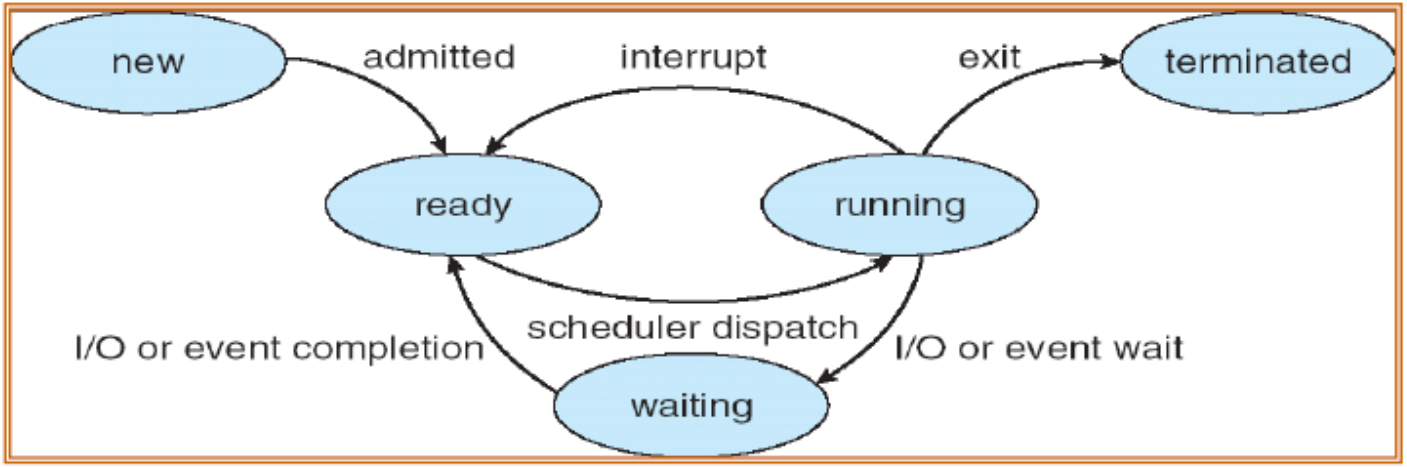
Masoud Baeimani

November 2019

الگوریتم های زمان بندی CPU
CPU Scheduling

زمانبند CPU

- انتخاب یکی از پردازه های آماده اجرا از حافظه و اختصاص پردازنده به آن
- تصمیم زمانبند در حالات زیر اتفاق می افتد:
  - تغییر وضعیت یک پردازه از حالت اجرا به انتظار
  - تغییر وضعیت یک پردازه از حالت اجرا به آماده
  - تغییر وضعیت یک پردازه از حالت انتظار به آماده
  - پایان یک پردازه



معیارهای زمانبند کوتاه مدت (پردازنده)

- بهره وری پردازنده(CPU utilization): مشغول نگه داشتن پردازنده تا حد توان
- توان عملیاتی(Throughput): تعداد پردازه های تکمیل شده در واحد زمان
- زمان برگشت(Turnaround time): زمان لازم برای تکمیل شدن یک پردازه خاص
  زمان برگشت: زمان I/O + زمان انتظار در صف آماده + زمان تعویض بستر + زمان اجرا
- زمان انتظار (Waiting time): مجموع زمانهایی که یک پردازه در صف انتظار بوده
- زمان پاسخ (Response time): مدت زمانی انتظار از لحظه تحویل یک پردازه تا در یافت اولین پاسخ

معیارهای بهینه سازی

- Max CPU utilization
- Max throughput
- Min turnaround time
- Min waiting time
- Min response time

حالت تصمیم

- بدون قبضه کردن (انحصاری)(Non Preemptive)
  - یک بار که فرآیند به حالت اجرا می رود، تا زمانی که کارش پایان یابد یا به حالت مسدود برود، کارش ادامه می یابد.

- باقبضه کردن (غیر انحصاری) (Preemptive)
  - فرآیند هایی که در حال حاضر در حال اجرا هستند، می توانند با بروز وقفه توسط سیستم عامل به حالت آماده بروند.

- با جلوگیری از به انحصار در امدن پردازنده توسط یک فرآیند به مدت طولانی، خدمات بهتری را ارائه می دهد.

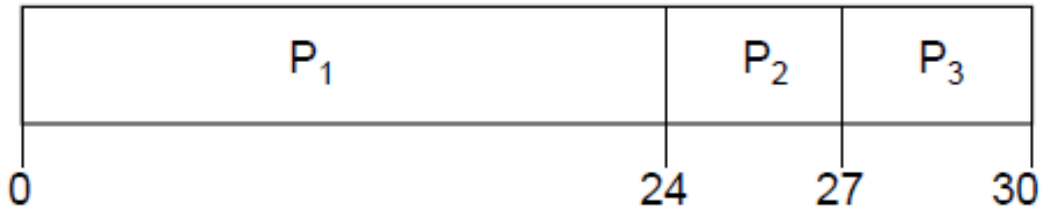
الگوریتم های زمانبندی معروف

- ✓ زمان‌بندی اجرا به ترتیب ورود (First-Come, First-Served)یا به اختصار(FCFS)
- ✓ زمان‌بندی کوتاه‌ترین کار بعدی (Shortest-Job-Next) یا به اختصار (SJN)
- ✓ زمان‌بندی اولویت (Priority)
- ✓ زمان‌بندی کوتاه‌ترین زمان باقیمانده (Shortest Remaining Time)
- ✓ زمان‌بندی نوبت گردشی یا راند رابین (Round Robin) یا به اختصار (RR)
- ✓ زمان‌بندی صف‌های چند سطحی (Multiple-Level Queues)

First-Come, First-Served (FCFS) زمانبندی

Process	Burst Time
P_1	24
P_2	3
P_3	3

- با فرض اینکه ترتیب ورود پردازه ها به صورت ذکر شده باشد.
- The Gantt Chart :



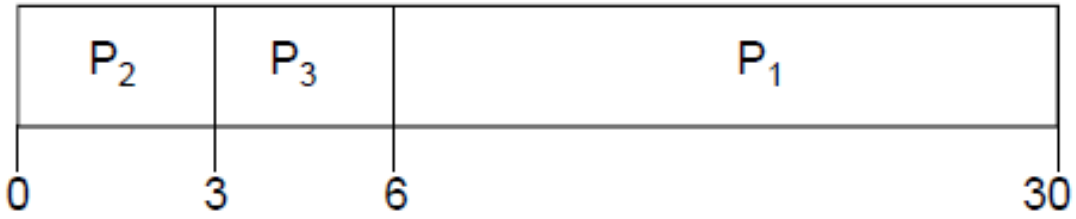
- Waiting time for $P_1 = 0$; $P_2 = 24$; $P_3 = 27$
- Average waiting time: $(0 + 24 + 27)/3 = 17$

زمانبندی FCFS

با فرض ترتیب ورود به صورت زیر:

$$P_2 , P_3 , P_1$$

- The Gantt chart :



- Waiting time for $P_1 = 6; P_2 = 0, P_3 = 3$
- Average waiting time: $(6 + 0 + 3)/3 = 3$

خدمت به ترتیب ورود(FCFS)

- به نفع پردازه های مقید به پردازنده(CPU Bound)، عمل می کند:
  - پردازه های مقید به ورودی-خروجی مجبورند تا زمانی که پردازه های مقید به پردازنده کامل شوند، منتظر بمانند.

- مزایا :
  - سادگی اجرا و عملی بودن

- معایب:
- در این حالت ممکن است که یک پردازه کوتاه زمان زیادی را در انتظار باشد.
- بالا بودن زمان برگشت یا زمان انتظار
- عدم امکان استفاده در سیستم های اشتراک زمانی
- انحصاری

زمانبندی SJF (Shortest Job First)

- به هر پردازه مدت زمان پردازش بعدی (CPU Burst) به پردازه ها اختصاص داده شده و از روی آنها هر یک کوتاهتر باشد انتخاب می شود.

- در دو صورت پیاده سازی شده است:
  - بدون قبضه کردن (non preemptive) انحصاری
  - با قبضه کردن (preemptive) غیر انحصاری

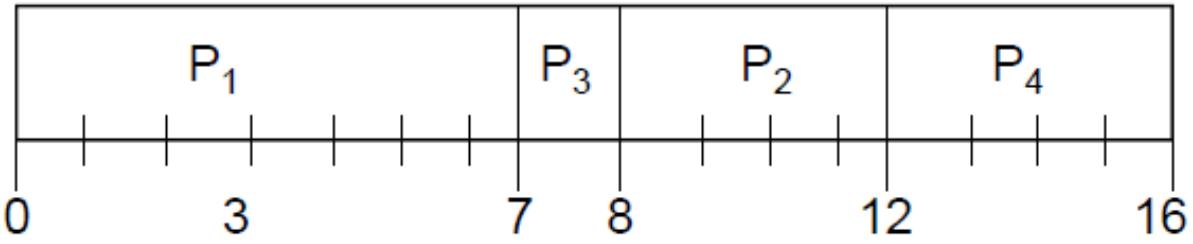
در این حالت اگر در حین اجرای پردازه، پردازه ای وارد شود که زمان پردازش لازم آن کمتر از مان باقی مانده پردازه در حال اجرا باشد، برای اجرا روی پردازنده انتخاب می شود.

- حالت با قبضه کردن به زمانبندی SRTF(Shortest-Remaining-Time-First) نیز مشهور است.

Example of SJF

Burst Time	Arrival Time	Process
7	0.0	P_1
4	2.0	P_2
1	4.0	P_3
4	5.0	P_4

- SJF (non-preemptive)

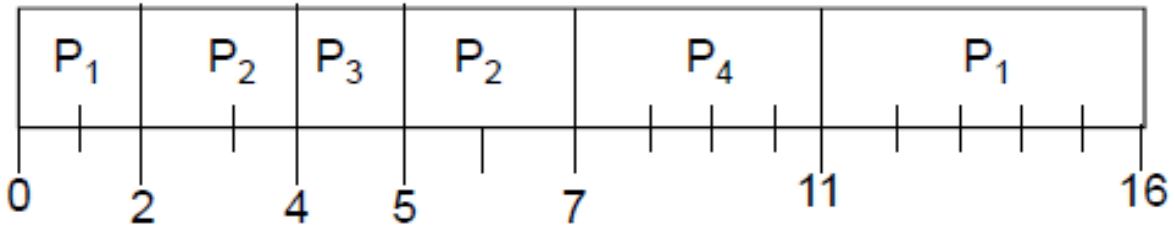


- Average waiting time = (0 + 6 + 3 + 7)/4 = 4

Example of SRTF

Burst Time	Arrival Time	Process
7	0.0	P_1
4	2.0	P_2
1	4.0	P_3
4	5.0	P_4

- SJF (preemptive)



- Average waiting time = (9 + 1 + 0 +2)/4 = 3

زمانبندی SJF (Shortest Job First)

- مزایا:
  - الگوریتم SJF بهینه بوده و کمترین میانگین زمان انتظار را برای یک مجموعه از پردازه ها را ایجاد می کند.

- معایب:
  - نیاز به داشتن اطلاعات مقاطع زمانی مورد نیاز قبل از شروع اجرا
  - احتمال گرسنگی (در حالت انحصاری)

زمانبندی اولویت(Priority)

- یک عدد اولویت به هر پردازه اختصاص داده شده می شود.
- پردازنده به پردازه ای اختصاص داده می شود که بالاترین اولویت را دارد
  (کوچکترین عدد = بالاترین اولویت)
- می تواند به صورت انحصاری یا غیر انحصاری پیاده سازی شود.
- زمانبندی SJF را می توان یک نوع زمانبندی اولویت فرض کرد که در آن عدد اولویت همان زمان پردازش بعدی پردازه است.
- مشکل
  - گرسنگی (Starvation) پردازه های با اولویت کم ممکن است هرگز اجرا نشوند
- راه حل
  - سالمندی (Aging) گذشت زمان به اولویت پردازه های در حال انتظار می افزاید

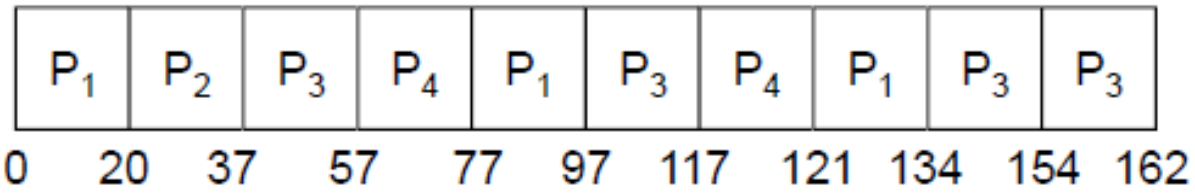
زمانبندی نوبت گردشی Round Robin (RR)

- پردازه ها به صورت دوره ای، برای زمان کوتاهی (کوانتوم زمانی) روی پردازنده اجرا می شوند.
- کوانتوم زمانی معمولا بین ۱۰ الی ۱۰۰ میلی ثانیه است و بعد از این زمان پردازه در حال به صف آماده منتقل می شود.
- اگر n پردازه وجود داشته باشد و کوانتوم زمانی q باشد
  - هیچ پردازه ای بیش از $(n-1)q$ *واحد زمانی منتظر نمی ماند*
- کارایی
  - برای کوانتوم های زمانی بسیار بزرگ شبیه FCFS عمل می کند
  - برای کوانتم های بسیار کوچک نیز سربار بسیار بالا است.
  - کوانتوم زمانی بایستی با توجه به زمان لازم برای تعویض بستر انتخاب شود.

Example of RR with Time Quantum = 20

Burst Time	Process
53	P_1
17	P_2
68	P_3
24	P_4

- The Gantt chart is:



- الگوریتم RR معمولا میانگین زمان تکمیل شدن بالاتری از SJF دارد ولی پاسخ بهتری دارد.

بالاترین نسبت پاسخ(HRRN)

- فرآیند بعدی را بر اساس فرمول زیر انتخاب می کند.

$$\frac{\textit{time spent waiting} + \textit{expected service time}}{\textit{expected service time}}$$

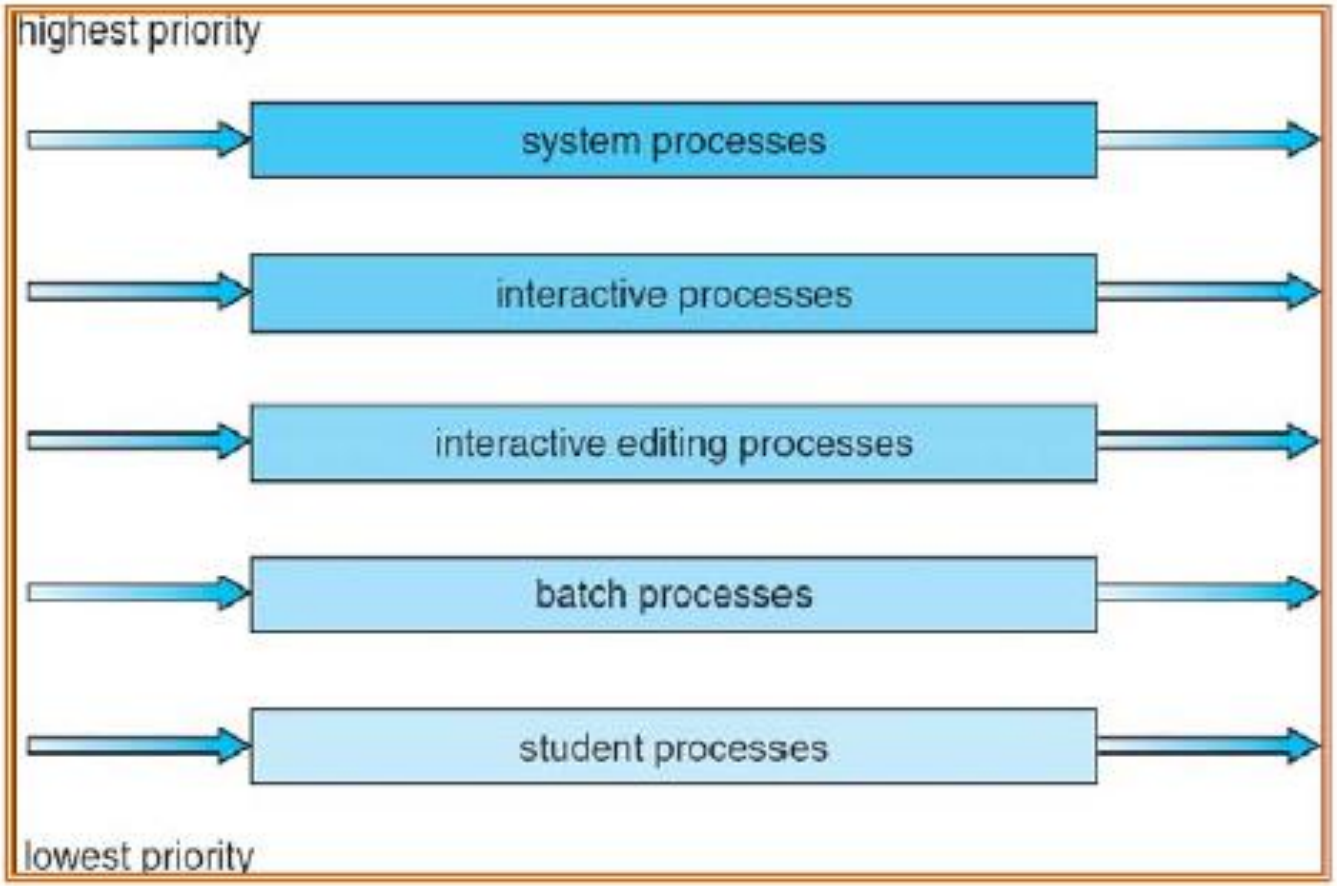
زمانبندی صف چندسطحی (Multilevel Queue)

- صف آماده به چند صف مجزا تقسیم بندی می شود.
  - پردازه های محاوره ای foreground (interactive)
  - پردازه های دسته ای background (batch)

- هر صف الگوریتم زمانبندی خاص خود را دارد
  - foreground – RR
  - background – FCFS

- بایستی بین صف ها نیز زمانبندی صورت گیرد.
  - ممکن است زمانبندی با اولویت ثابت برای صف ها در نظر گرفته شود(مثلا همه پردازها در صف محاوره ای باید سرویس داده شود تا نوبت پردازه های صف دیگر برسد) امکان گرسنگی در این حالت وجود دارد
  - استفاده از برش زمانی: هر صف مقدار خاصی از زمان پردازنده را در اختیار گرفته و با الگوریتم خاص خود بین پردازه هایش تقسیم می کند. مثلا ۸۰٪ از برش زمانی برای صف محاوره ای و ۲۰٪ برای صف دیگر.

زمانبندی صف چندسطحی (Multilevel Queue)



زمانبندی صف چند سطحی بازخورد (Multilevel Feedback Queues)

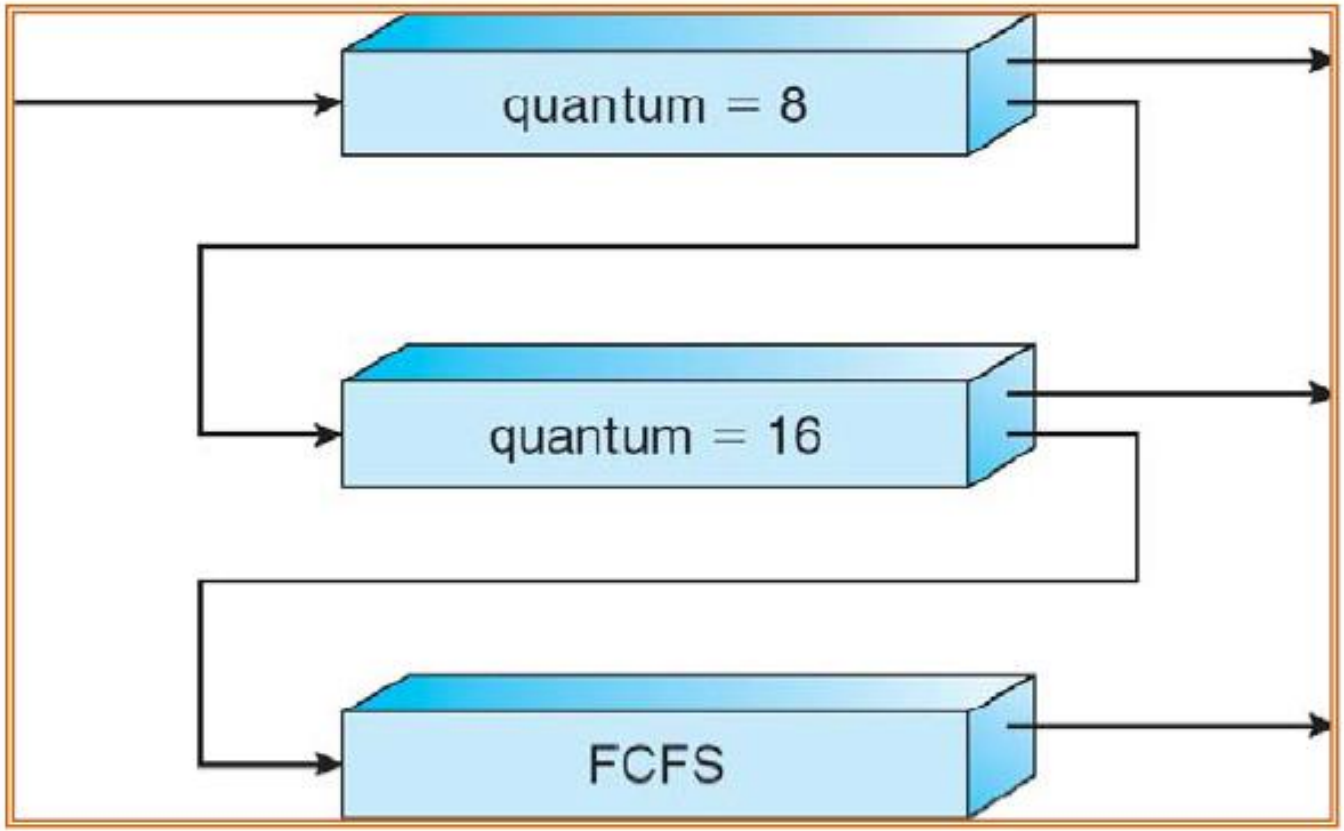
- یک پردازه می تواند بین صف ها حکرت کند (الگوریتم سالمندی به این منظور می تواند به کار برده شود)

- زمانبندی صف چندسطحی بازخورد بر اساس معیارهای زیر می باشد:
  - تعداد صف ها
  - الگوریتم زمانبندی هر یک از صف ها
  - متد معین کننده زمان افزایش اولویت و تغییر صف یک پردازه
  - متد معین کننده زمان کاهش اولویت و تغییر صف یک پردازه
  - متدی که در هنگام ورود پردازه صف مناسب را مشخص می کند

مثال MFQ

- سه صف:
  - Q_0 : RR با کوانتوم زمانی 8 میلی ثانیه
  - Q_1 : RR با کوانتوم زمانی 16 میلی ثانیه
  - Q_2 : FCFS
- زمانبندی:
- پردازه جدید به Q_0 وارد شده و 8 میلی ثانیه دریافت می کند و در صورتیکه زمان پردازش آن تمام نشود به صف Q_1 منتقل می شود و به در نوبت خود 16 میلی ثانیه دیگر نیز دریافت می کند و در صورتیکه تکمیل نشود به Q_2 منتقل می شود..Q_2

Multilevel Feedback Queues



زمانبندی چند پردازنده ای

- زمانبندی پردازنده زمانیکه چند پردازنده موجود باشد، پیچیده تر خواهد بود.
- حل مسئله *Load sharing*
- چند پردازنده ای متقارن (Symmetric Multiprocessing): هر پردازنده زمانبندی خود را انجام میدهد.
- چند پردازنده ای نا متقارن (*Asymmetric multiprocessing*): یک پردازنده مسئولیت زمانبندی را بر عهده می گیرد.

زمانبندی بلادرنگ(Real-Time)

- سیستم های بلادرنگ سخت (Hard real-time)

بایستی وظیفه بحرانی محول شده در یک زمان تضمین شده پایان یابد

- محاسبات بلادرنگ نرم (Soft real-time computing)

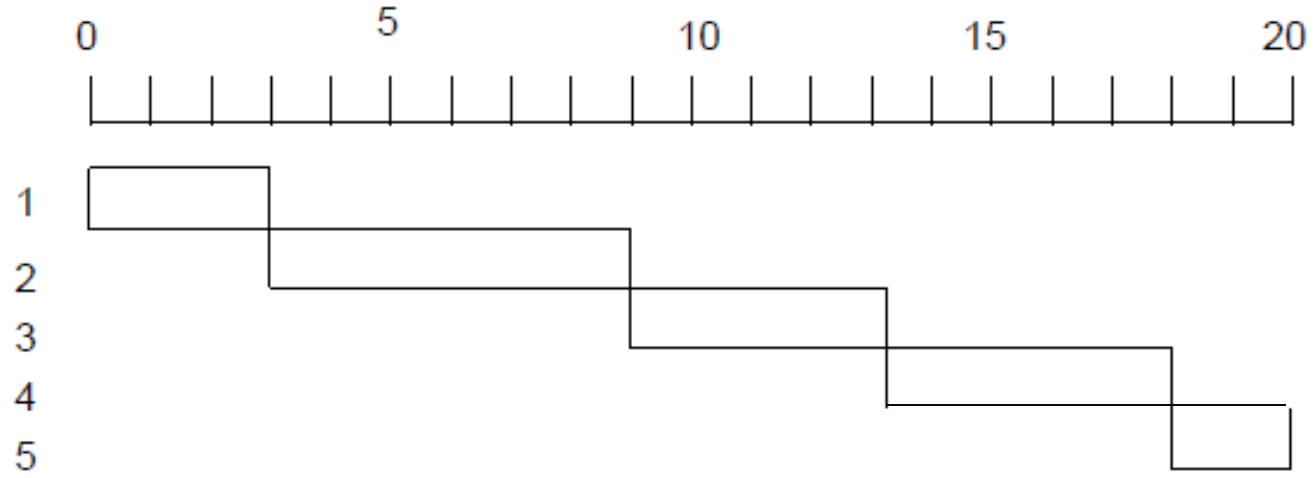
برای پردازه های حیاتی اولویت بالاتری در نظر گرفته می شود.

مثال زمانبندی فرآیند

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

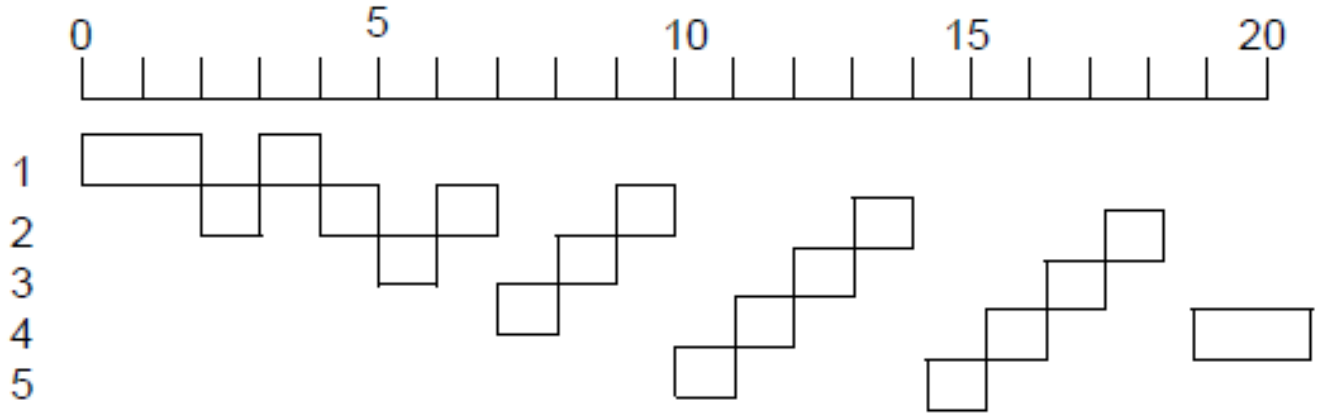
خدمت به ترتیب ورود(FCFS)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



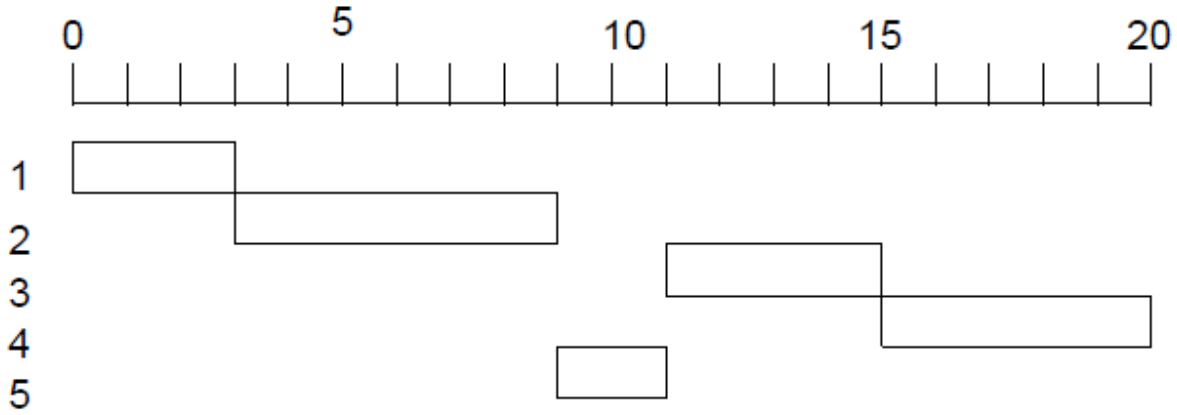
نوبت گردشی (RR)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



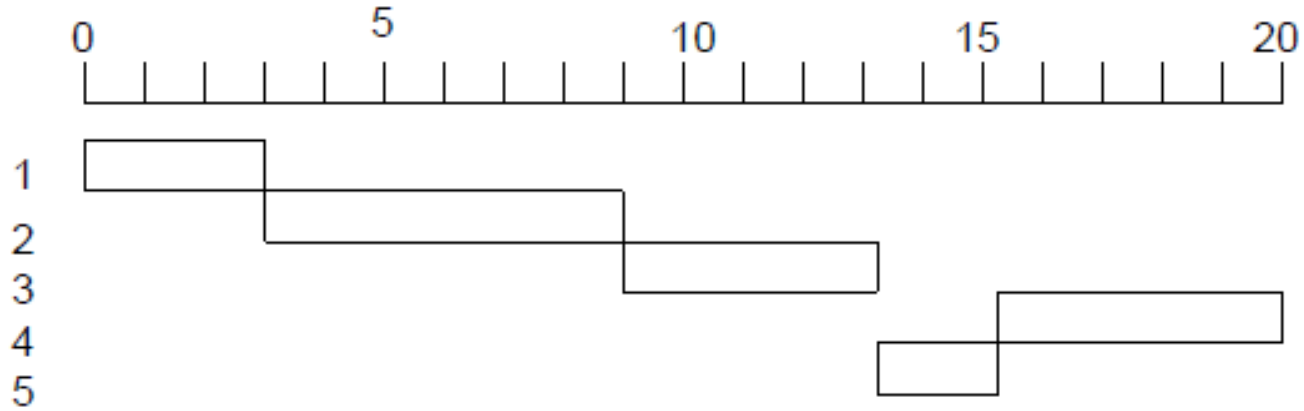
کوتاه ترین فرآیند (SJF)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



بالاترین نسبت پاسخ(HRRN)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



زمانبندی MFQ

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2