

الف) در صورت زیاد بودن این برهه زمانی، پروسه هایی با cpu burst های بزرگ تر نفع بیشتری نسبت به دیگر پروسه ها میبرند. برهه های زمانی کوتاه برای سیستم هایی که کانتکست سویچ سریعی دارند مناسب تر است. هر چند کوانتوم باید نسبت به سویچ بسیار طولانی تر باشد.

ب) مشابه FIFO عمل میکند و مناسب تر تسک های کوتاه تر است، همینطور overhead کمتری دارد

ج) کوانتوم طولانی مناسب تسک های کوتاه تر است، و کوانتوم کوتاه تر مناسب هر نوع تسک

از نظر cpu utilization کوانتوم بلند تر بهتر است، زیرا overhead کمتری دارد

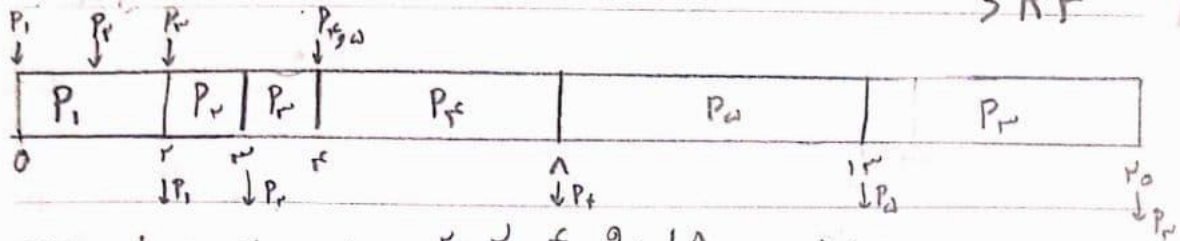
کوانتوم کوتاه responsive تر است و کوانتوم بلند throughput بیشتری دارد

د) برای سیستم هایی که چند نوع تسک (batch, interactive) دارند، ترکیب این دو مناسب تر است، به این صورت که یک صف با کوانتوم طولانی تر برای batch و یک صف با کوانتوم کوتاه تر برای interactive ها در نظر گرفته میشود

Year Month Date ()

Subject

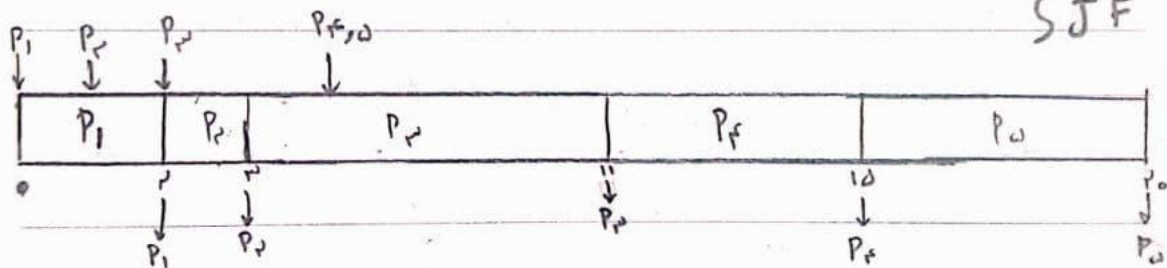
SRF



$$\text{avg turnaround: } \frac{1+2+3+4+5+6}{6} = V$$

$$\text{avg waiting: } \frac{0+1+2+3+4+5}{6} = W$$

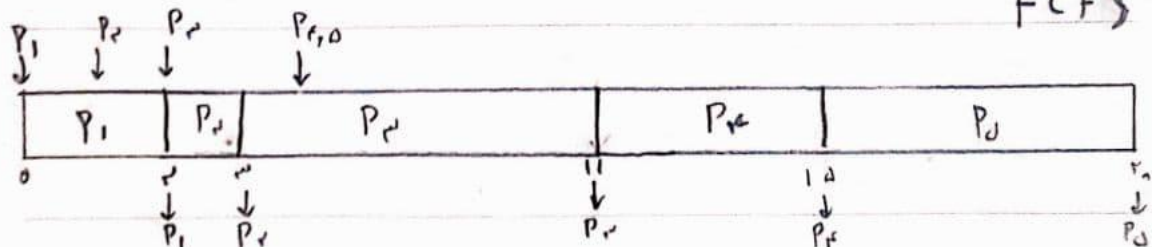
SJF



$$\text{avg Turnaround: } \frac{1+2+3+4+5+6}{6} = \Lambda$$

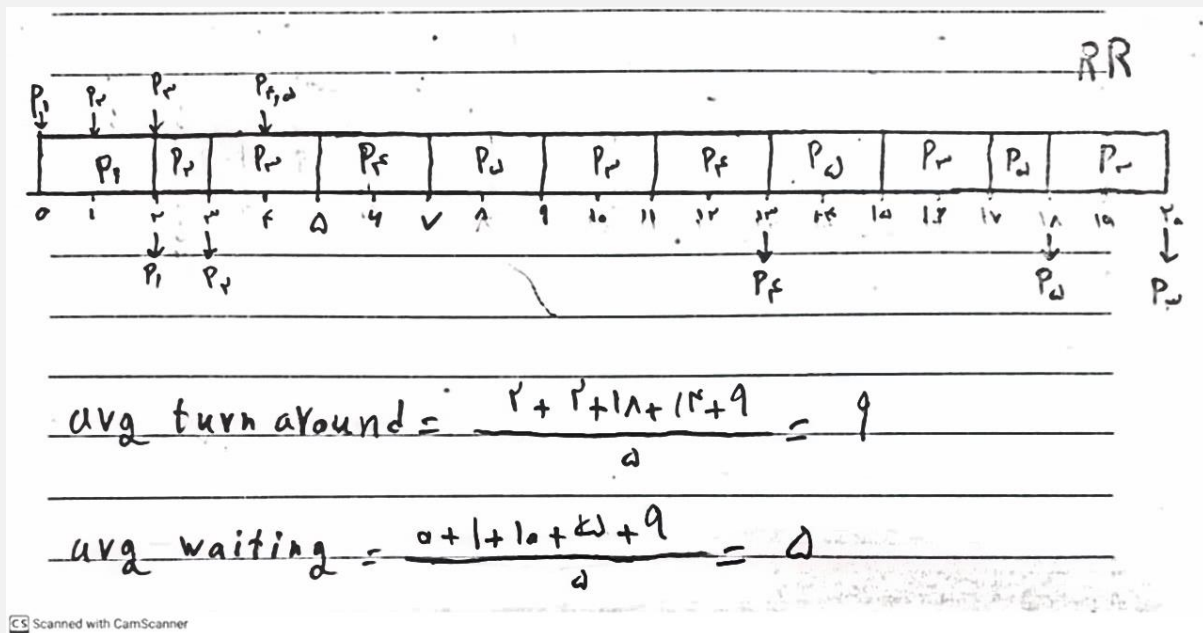
$$\text{avg waiting: } \frac{0+1+2+3+4+5}{6} = F$$

FCFS



$$\text{avg Turnaround} = \Lambda$$

$$\text{avg, waiting} = F$$



-3

(الف) پروسه های CPU-bound بیشتر سود میکنند، چون هم اولویتشان افزایش میابد، هم waiting time و Turnaround کمتری دارند، به طور کل سریع تر به هدف میرسند

(ب) کاهش overhead و افزایش cpu utilization

-4

(الف) ماژولی است که کنترل سی پی یو را به پروسه انتخاب شده توسط scheduler میدهد، به طور کل وظایف context switch, user mode switch و پرش به آدرس مناسب برای ادامه اجاره پروسه را به عهده دارد

(ب) هنگام اجرای یک thread در یک پردازنده، حافظه های مورد استفاده آن داخل کش پردازنده ذخیره میشوند، به این حالت وابستگی به پردازنده گفته میشود

(ج) اشاره به رابطه ران تایم و اولویت یک پروسه دارد، هر چه اولویت یک پروسه بالاتر باشد، runtime آن کمتر از ران تایم واقعی آن است

-5

الف) زمانی که چندین پروسه با برست های طولانی به صورت یک جا وارد صف میشوند، اگر بخواهیم response time را پایین نگه داریم، به دلیل کانتکست سویچ های زیاد و overhead آن، cpu utilization کاهش میابد

ب)

-6

الگوریتم های SJF و Priority

در الگوریتم Priority اگر یک پروسه اولویت خیلی پایینی داشته باشد، و مرتبا پروسه هایی با اولویت بالا وارد سیستم شود، پروسه اولویت پایین دچار گرسنگی میشود زیرا زمانبند تا زمانی که پروسه با اولویت بالا در صف است آن را انتخاب میکند

همینطور الگوریتم SJF یک الگوریتم اولویت دار است و پروسه هایی با برست کوتاه تر اولویت بالاتر دارد، در نتیجه میتواند برای پروسه ها با برست های طولانی تر اتفاق مشابهی رخ دهد

-7

الگوریتم FCFS، زمانی که یک پروسه کوچک در پشت چندین پروسه بزرگ قرار میگیرد ضرر میکند، چون باید تا اتمام کامل آن ها در صف بماند

الگوریتم multi-level feedback queues به نفع پروسه های کوچک تر عمل میکند، زیرا اگر یک پروسه برست های کوچک تر از کوانتوم صف اول داشته باشد، مرتبا در صف اولویت بالا قرار میگیرد