

فصل ۳:

زمان بندی پردازنده

هدف از زمان بندی پردازنده، تخصیص فرایندها به پردازنده در طول زمان است به گونه ای که هدف های سیستم از قبیل زمان پاسخ، توان عملیاتی و کارایی پردازنده را برآورده سازد. زمان بندی پردازنده، اساس سیستمهای عامل چند برنامه ای است. با حرکت پردازنده بین فرایندها، سیستم عامل می تواند بهره وری کامپیوتر را افزایش دهد.

معیارهای زمان بندی

معیارهای زمان بندی به دو دسته تقسیم می شوند. معیارهای از دید کاربر مانند زمان پاسخ و معیارهای از دید سیستم : مانند توان عملیاتی، بهره وری پردازنده.

توان عملیاتی (throughput) : تعداد فرایندهایی که در واحد زمان تکمیل می شوند.

زمان انتظار = " زمان خروج - زمان اجرا - زمان ورود "

زمان برگشت = " زمان خروج - زمان ورود "

زمان برگشت = " زمان انتظار + زمان اجرا "

تذکر: نام های دیگر زمان برگشت (turnaround time) عبارتند از :

زمان اجرای کامل، زمان کل، زمان تکمیل و زمان پاسخ (response time).

استفاده از پردازنده (بهره وری CPU) در سیستم های اشتراکی معیار مهمی است.

به زمان مشخصی از پردازنده که برای اجرای مجموعه دستورالعملهای مربوطه به CPU در نظر گرفته می شود، CBT (CPU Bust Time) می گویند.

از دید کاربر زمان پاسخ و از دید سیستم بهره وری پردازنده، مهم است.

معیارهای زمان بندی گفته شده به یکدیگر وابسته هستند و بهینه سازی همه آن ممکن نمی باشد.

دسته بندی سیاست های زمان بندی

۱- بدون قبضه کردن (non preemptive)

در حالت بدون قبضه کردن (انحصاری)، همین که یک فرایند در حالت اجرا قرار گرفت، آنقدر به اجرا ادامه می‌دهد تا خاتمه یابد یا اینکه خودش (داوطلبانه)، برای انتظار I/O مسدود شود.

۲- با قبضه کردن (preemptive)

در حالت با قبضه کردن (غیر انحصاری)، فرایند در حال اجرا می‌تواند توسط سیستم عامل متوقف شود و به حالت آماده منتقل شود.

تصمیم به قبضه کردن می‌تواند در یکی از حالات زیر انجام گیرد:

۱- به صورت دوره ای براساس وقفه ساعت

۲- در زمان ورود یک فرایند جدید

۳- در زمانی که فرایند مسدودی به حالت آماده برود.

سیاست هایی که با قبضه کردن همراه است سربرار بیشتری را به همراه دارند ولی خدمت بهتری را ارائه می‌دهند.

الگوریتم های زمان بندی

الگوریتم های زمان بندی پردازنده عبارتند از:

۱- سرویس به ترتیب ورود (FCFS)

۲- نوبت گردشی (RR)

۳- کوتاهترین فرایند (SJF یا SPN)

۴- کوتاهترین زمان باقی مانده (SRT)

۵- بالاترین نسبت پاسخ (HRRN)

۶- فیدبک (FB)

و الگوریتم های زیر که در کتاب سیلبرشاتز آورده شده است.

Priority - ۳ MLQ - ۲ MLFQ - ۱

FCFS : First-Come First-Served

RR : Round Robin

SPN : Shortest Process Next

SRT : Shortest Remaining Time

HRRN : Highest Response Ratio Next

FB : Feed Back

MLFQ : Multi Level Feedback Queue

MLQ : Multi Level Queue

سرویس به ترتیب ورود (FCFS)

در این الگوریتم فرایندی انتخاب می‌شود که بیشتر منتظر بوده است، یعنی زودتر CPU را درخواست کرده است. پیاده سازی این الگوریتم با یک صف (FIFO) انجام می‌شود. وقتی فرایندی وارد صف آماده می‌شود، PCB آن در انتهای صف قرار می‌گیرد و با آزاد شدن CPU، فرایند موجود در اول صف، CPU را در اختیار خواهد گرفت و از صف حذف می‌شود.

ویژگی های زمان بندی FCFS

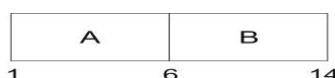
- ۱- گرسنگی ندارد.
 - ۲- برای فرایندهای طولانی بسیار بهتر از فرایندهای کوتاه عمل می‌کند.
 - ۳- سربار حداقل است، چون نیازی به اطلاعات قبلی در مورد فرایندها نمی‌باشد.
 - ۴- تابع انتخاب برابر Max (wait) است.
 - ۵- میانگین زمان انتظار بسیار بالا می‌باشد.
 - ۶- در یک سیستم تک پردازنده ای، روش خوبی نیست.
 - ۷- انحصاری است.
 - ۸- به فرایندهای I/O bound ضربه می‌زند. (چون وقتی فرایند CPU bound در حال اجراست، فرایندهای I/O bound باید منتظر باشند).
 - ۹- می‌تواند باعث استفاده ناکارآمد از CPU و دستگاههای I/O شود. وقتی فرایند جاری حالت اجرا را ترک می‌کند، فرایندهای آماده I/O bound به سرعت حالت اجرا را گذرانده و برای رویدادهای I/O مسدود می‌شوند. در این لحظه، اگر فرایند CPU bound نیز مسدود باشد، CPU بیکار می‌شود.
- تذکر: برای بدست آوردن زمان پاسخ و زمان انتظار فرایندها از روشی به نام گانت استفاده می‌شود.

مثال

با توجه الگوریتم FCFS، گانت را برای فرایندهای زیر رسم نمایید.

	زمان ورود	زمان اجرا
A	1	5
B	4	8

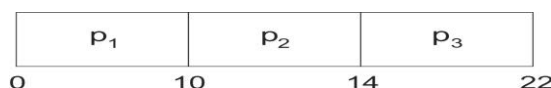
حل:



این نمودار مشخص می کند که در لحظه یک اجرای فرایند A ، بدون هیچ انتظاری شروع شده و بعد از 5 میلی ثانیه در لحظه 6 ، پایان یافته است. سپس اجرای فرایند B با 2 میلی ثانیه تاخیر شروع شده و بعد از 8 ثانیه دیگر در لحظه 14 پایان یافته است. از گانت می توان زمان پاسخ را نیز بدست آورد. کافی است که زمان ورود (که در جدول اولیه داده شده) را از زمان خروج (که در گانت مشخص است) کم کرد. بنابراین زمان پاسخ فرایند A برابر 5 میلی ثانیه (6-1) و زمان پاسخ فرایند B برابر 10 میلی ثانیه (14-4) می باشد. ■

مثال

در صورت استفاده از الگوریتم FCFS ، میانگین زمان انتظار برای سه فرایند P1 و P2 و P3 با زمانهای اجرای 10 و 4 و 8 میلی ثانیه را بدست آورید. (فرایندها در زمان صفر وارد شده اند).
حل: ابتدا فرایند P1 اجرا شده که 10 میلی ثانیه طول می کشد، سپس فرایند P2 که 10 میلی ثانیه منتظر بوده است اجرا شده و بعد از 4 میلی ثانیه (در زمان 14) اجرای آن تمام می شود. در نهایت فرایند P3 بعد از 14 میلی ثانیه انتظار اجرا می شود. گانت آن به صورت زیر است:



زمان انتظار را به کمک رابطه "زمان خروج - زمان اجرا" محاسبه می کنیم:

$$\frac{(10-0) + (14-10) + (22-14)}{3} = \frac{0+10+14}{3} = 8$$

■

مثال

در الگوریتم FCFS با توجه به فرایندهای داده شده، مطلوب است:

الف- میانگین زمان اجرا

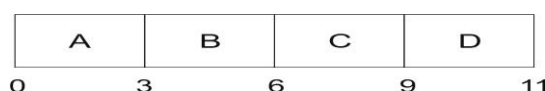
ب- میانگین زمان انتظار

ج- میانگین زمان برگشت (پاسخ)

نام فرایند	زمان ورود	زمان پردازش
A	0	3
B	1	3
C	4	3
D	6	2

حل:

گانت فرایندها به صورت زیر می باشد:



الف- میانگین زمان اجرا:

$$\frac{3+3+3+2}{4} = \frac{11}{4}$$

ب- میانگین زمان انتظار:

$$\frac{(3-3-0) + (6-3-1) + (9-3-4) + (11-2-6)}{4} = \frac{0+2+2+3}{4} = \frac{7}{4}$$

ج- میانگین زمان برگشت:

$$\frac{(3-0) + (6-1) + (9-4) + (11-6)}{4} = \frac{3+5+5+5}{4} = \frac{18}{4}$$

تذکر: البته می‌توانستیم با توجه به رابطه زیر نیز، میانگین زمان برگشت را محاسبه کنیم:

میانگین زمان برگشت = میانگین زمان اجرا + میانگین زمان انتظار



نوبت گردشی (RR)

این الگوریتم شبیه به FCFS است، با این تفاوت که زمانبند پردازنده بین فرایندها در یک صف چرخشی حرکت کرده و CPU حداکثر به مدت یک کوانتوم زمانی به هر فرایند تخصیص داده می‌شود. اگر در یک کوانتوم زمانی، اجرای پروسسی تمام نشود به ته صف می‌رود.

ویژگی‌های روش RR

- ۱- در سیستمهای اشتراک زمانی یا در سیستم پردازش تراکنش بسیار مؤثر می‌باشد.
- ۲- اگر برهه زمانی از زمان اجرای بلندترین فرایند بیشتر باشد، سیاست RR به FCFS تنزل می‌یابد.
- ۳- زمان بندی RR، غیر انحصاری (قبضه شدنی) می‌باشد.
- ۴- گرسنگی ندارد.
- ۵- عملکرد آن عادلانه است.
- ۶- اگر برهه زمانی خیلی کوچک باشد، توان عملیاتی آن کم است.
- ۷- برای فرایندهای کوتاه، زمان برگشت خوبی ارائه می‌کند.
- ۸- از مشکلات آن، رفتار با فرایندهای در تنگنای پردازنده در مقایسه با فرایندهای در تنگنای I/O می‌باشد.

مثال

میانگین زمان انتظار سه پردازش زیر با استفاده از سیاست زمان بندی RR با کوانتوم زمانی 3 میلی ثانیه را بدست آورید؟ (زمان ورود = صفر) (زمان پردازش : $P_1=9, P_2=2, P_3=2$)
حل: گانت بصورت زیر می‌باشد:

p_1	p_2	p_3	p_1	p_1	
0	3	5	7	10	13

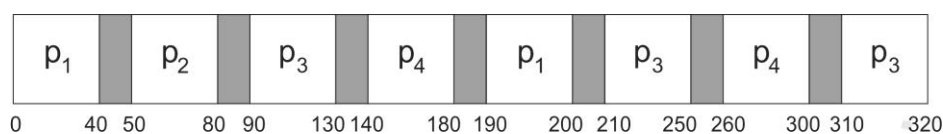
اجرای پروسس P1 در اولین برش زمانی تمام نمی‌شود و به انتهای صف می‌رود و پروسس P2 و P3 قبل از پایان برش زمانی، به پایان می‌رسند. میانگین زمان انتظار برابر است با:

$$\frac{(13-9) + (5-2) + (7-2)}{3} = 4$$



مثال

اگر زمان اجرای پروسسهای P1, P2, P3, P4 به ترتیب 50, 30, 90, 80 باشد و روش RR با کوانتوم زمانی 40 و زمان تعویض متن 10 استفاده شود، میانگین زمان برگشت کدام است؟ (زمان ورود = 0)
حل: گانت فرایندها به صورت زیر است:



میانگین زمان برگشت برابر است با:

$$\frac{200 + 80 + 320 + 300}{4} = 225$$

مثال

پنج فرایند با مشخصات زیر به یک سیستم با زمان بندی RR با برش زمانی q=1 وارد شوند. نمودار گانت را رسم کنید.

	زمان ورود	زمان اجرا
P1	0	2
P2	0	2
P3	1	1
P4	1	1
P5	2	1

فرض: همیشه بین فرایندی که در لحظه t برش زمانی خود را به پایان می‌رساند و فرایند ورودی در لحظه t، اولویت با فرایند قبلی موجود در سیستم است و در شرایط کاملاً یکسان بین دو فرایند، اولویت با فرایند با شماره کوچکتر است.

حل:

گانت به صورت زیر می‌باشد:

P ₁	P ₂	P ₁	P ₃	P ₄	P ₂	P ₅
0	2	3	4	5	6	7
1						

در لحظه 0، P1 و P2 وارد می شوند. پردازنده به P1 داده می شود و P2 در صف قرار می گیرد. در لحظه 1، پردازنده از P1 گرفته شده و بعد از P2 در صف قرار می گیرد. همچنین P3 و P4 که در همین لحظه وارد شده اند، در صف پشت سر P1 قرار می گیرند. پردازنده به P2 داده می شود. در لحظه 2، پردازنده از P2 گرفته شده و به انتهای صف بعد از P4 رفته و P5 که در همین لحظه وارد شده در انتهای صف بعد از P2 قرار می گیرد. پردازنده هم به P1 (فرایند ابتدای صف) داده می شود. در لحظه 3، اجرای P1 تمام شده و پردازنده به P3 که اول صف است داده می شود. در لحظه 4، اجرای P3 تمام شده و پردازنده به فرایند اول صف یعنی P4 داده می شود. در لحظه 5، اجرای P4 تمام شده و پردازنده به P2 داده می شود. در لحظه 6، اجرای P2 تمام شده و پردازنده به P5 داده می شود.

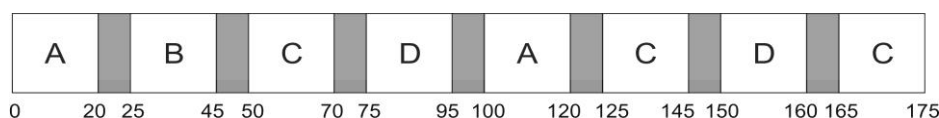


مثال

به کمک روش زمان بندی نوبه‌ای با کوانتوم زمانی 20 میلی ثانیه، متوسط زمان انتظار پردازنده‌ها را محاسبه کنید؟ (زمان Context Switch برابر 5 میلی ثانیه است). (زمان ورود همه فرایندها = 0)

فرایند	زمان اجرا
A	40
B	20
C	50
D	30

حل: گانت آن به صورت زیر است:



و میانگین زمان انتظار برابر است با :

$$\frac{(120 - 40) + (45 - 20) + (175 - 50) + (160 - 30)}{4} = \frac{80 + 25 + 125 + 130}{4} = 90$$

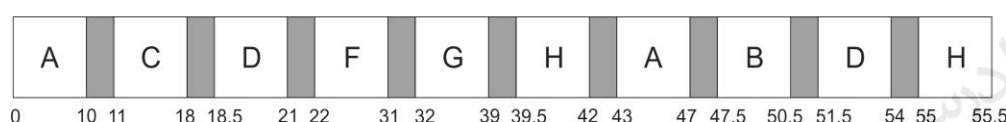


مثال

سیستمی شامل چهار فرایند است که داخل هر فرایند می تواند بیش از یک نخ اجرایی وجود داشته باشد مطابق جدول زیر مفروض است. برای فرایندها از الگوریتم Round-Robin با برش زمانی $q=10$ استفاده کنید. داخل هر فرایند از روش FIFO برای تعویض نخ ها استفاده می شود و تا زمانی که اجرای یک نخ تمام نشده، نوبت به نخ بعدی نمی رسد. برای تعویض فرایند 1ms و برای تعویض نخ در داخل فرایند 0.5ms زمان لازم است. نمودار گانت را رسم کنید.

فرایند	P ₁		P ₂		P ₃	P ₄	
نخ	A	B	C	D	F	G	H
زمان اجرا	14	3	7	5	9	7	3

حل: نمودار گانت به صورت زیر می باشد:

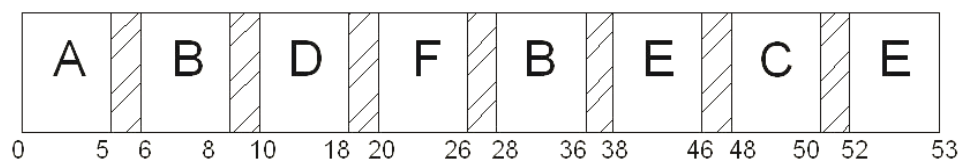


مثال

سیستمی شامل سه فرایند مطابق جدول زیر مفروض است. برای فرایندها از الگوریتم Round-Robin با برش زمانی 8 میلی ثانیه استفاده کنید. داخل هر فرایند از روش FIFO برای تعویض نخ ها استفاده می شود و تا زمانی که اجرای یک نخ تمام نشده، نوبت به نخ بعدی نمی رسد. برای تعویض فرایند 2ms و برای تعویض نخ در داخل فرایند 1ms زمان لازم است. میانگین زمان پاسخ را برای نخ های هر فرایند محاسبه کنید.

فرایند	P ₁			P ₂		P ₃
نخ	A	B	C	D	E	F
زمان اجرا	5	10	2	8	9	6

حل: نمودار گانت به صورت زیر می باشد:



میانگین زمان پاسخ برای نخ‌های هر فرایند برابر است با:

$$A = 5, B = 36, C = 50 \Rightarrow \frac{91}{3}$$

$$D = 18, E = 53 \Rightarrow \frac{71}{2}$$

$$F = 26$$



کوتاهترین فرایند (SPN یا SJF)

در این سیاست فرایندی برای اجرا انتخاب می‌شود که به کوتاهترین زمان پردازش نیاز دارد. یعنی فرایند کوتاه از روی فرایندهای بلند می‌گذرد و به ابتدای صف می‌آید.

ویژگی‌های الگوریتم SPN

۱- از معایب آن، نیاز به دانستن زمان پردازش هر فرایند است.

۲- انحصاری است.

۳- امکان گرسنگی فرایندهای طولانی وجود دارد.

۴- برای محیط‌های اشتراک زمانی، مناسب نیست.

۵- میانگین زمان انتظار، کمینه است.

مثال

میانگین زمان انتظار را برای فرایندهای زیر به روش SJF بدست آورید؟ (ورود همه در لحظه صفر)
($P_1=7$, $P_2=8$, $P_3=3$, $P_4=5$)

حل: ترتیب اجرا برابر است با: (از کوتاهترین کار شروع می‌شود)

p_3	p_4	p_1	p_2	
0	3	8	15	23

میانگین زمان انتظار برابر است با:

$$\frac{(15-7) + (23-8) + (3-3) + (8-5)}{4} = 6.5$$

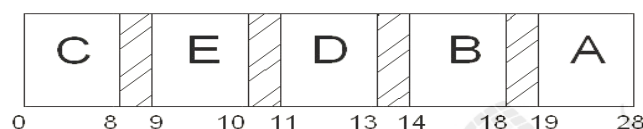


مثال

با توجه به جدول زیر متوسط زمان انتظار را در هر یک از روشهای SJF محاسبه کنید.
(زمان تعویض متن = ۱ میلی ثانیه)

پروسس	زمان ورود به سیستم	زمان مورد نیاز پردازش
A	0	9
B	2	4
C	0	8
D	3	2
E	5	1

پاسخ: نمودار گانت به صورت زیر است:



زمان انتظار هر یک از فرایندها برابر است با:

$$A = 19, B = 12, C = 0, D = 8, E = 4 \Rightarrow \frac{19 + 12 + 0 + 8 + 4}{5} = \frac{43}{5}$$

پیش بینی زمان انفجار محاسباتی بعدی

در الگوریتم SJF راهی وجود ندارد که از انفجار محاسباتی بعدی آگاهی پیدا کنیم. یک روش این است که زمانبندی SJF تخمین زده شود. ممکن است طول بعدی را ندانیم، اما می توانیم اندازه اش را پیش بینی کنیم. با تخمین طول انفجار محاسباتی بعدی، می توانیم فرآیندی را انتخاب کنیم که طول انفجار محاسباتی بعدی آن کوتاهتر است. برای تخمین از رابطه $S_n = \alpha S_{n-1} + (1 - \alpha)T_{n-1}$ استفاده می شود. ($0 \leq \alpha \leq 1$)

مثال

سه وظیفه A, B, C را در نظر بگیرید که تاکنون n-1 بار در سیکل آماده-اجرا-مسدود طی مسیر کرده اند. زمان اجرای واقعی سیکل n-1 ام این وظایف بترتیب 2، 4 و 6 میلی ثانیه و زمان برآورد شده برای اجرای n-1 ام آنها نیز بترتیب 4، 6 و 6 میلی ثانیه می باشد. زمان اجرای واقعی در سیکل n ام به ترتیب 5، 4 و 7 است. با فرض $\alpha = 0.5$ نحوه زمانبندی این وظایف با استفاده از الگوریتم SJF را مشخص کنید.

حل: توسط الگوریتم سالمندی (aging)، می‌توان زمان اجرای کارها را به کمک رابطه $S_n = \alpha S_{n-1} + (1 - \alpha)T_{n-1}$ تخمین زد. که S_{n-1} زمان برآورد مرحله قبل و T_{n-1} زمان واقعی مرحله قبل می‌باشد. با فرض $\alpha = 0.5$ داریم:

$$S_n = 0.5 \times S_{n-1} + (1 - 0.5) \times T_{n-1} = 0.5 \times (S_{n-1} + T_{n-1})$$

بنابراین می‌توان S_n را به صورت زیر محاسبه کرد:

	S_{n-1}	T_{n-1}	S_n
A	4	2	$\frac{1}{2} \times (4 + 2) = 3$
B	6	4	$\frac{1}{2} (6 + 4) = 5$
C	6	6	$\frac{1}{2} (6 + 6) = 6$

با توجه به S_n به دست آمده، مشخص است که ترتیب اجرا برابر است با: $A \rightarrow B \rightarrow C$

(طبق الگوریتم SJF، کارها از کوچک به بزرگ اجرا می‌شوند)

حال با توجه به زمان اجرای واقعی در سیکل n ام که برابر 5، 4 و 7 می‌باشد، نمودار گانت را رسم می‌کنیم:

A	B	C	
0	5	9	16

کوتاهترین زمان باقیمانده (SRT)

زمان بندی SRT یک نوع SPN با قبضه کردن است و فرایندی برای اجرا انتخاب می‌شود که انتظار می‌رود کوتاهترین زمان پردازش باقیمانده را داشته باشد. اگر فرایند جدیدی وارد صف آماده شود و زمان باقیمانده کمتری نسبت به فرایندی که در حال اجراست داشته باشد، فرایند در حال اجرا قبضه می‌شود و فرایند جدید اجرا می‌شود.

ویژگی های الگوریتم SRT

- ۱- غیر انحصاری است.
- ۲- امکان گرسنگی برای کارهای طولانی زیاد است.
- ۳- بر خلاف RR، وقفه های اضافی بوجود نمی‌آید، بنابراین سربار کاهش می‌یابد ولی از طرف دیگر، زمان خدمت سپری شده باید ثبت شود، که ایجاد سربار می‌کند.
- ۴- زمان کل SRT نسبت به SPN بهتر است، چون کار کوتاه اولویت بیشتری نسبت به کار بلند در حال اجرا دارد.

مثال

در صورت استفاده از الگوریتم SRT برای فرایندهای زیر، میانگین زمان انتظار را بدست آورید.

فرایند	زمان ورود	زمان اجرا
P1	0	8
P2	1	4
P3	2	9
P4	3	5

حل: اجرای فرایند P1 در لحظه صفر شروع شده و با ورود P2 در زمان 1، اجرای P1 قطع شده (چون P2 زمان اجرای کمتری نسبت به زمان باقیمانده برای P1 یعنی 7 میلی ثانیه دارد) و اجرای P2 شروع خواهد شد و بعد از اجرای P2، اجرای P4 شروع می‌شود و سپس اجرای P1 ادامه می‌یابد و در نهایت P3 اجرا خواهد شد. بنابراین گانت آن به صورت زیر است:

p ₁	p ₂	p ₄	p ₁	p ₃	
0	1	5	10	17	26

میانگین زمان انتظار برابر است با:

$$\frac{(17-8-0) + (5-4-1) + (26-9-2) + (10-5-3)}{4} = 6.5$$

مثال

با توجه به جدول زیر متوسط زمان انتظار را در هر یک از روشهای SRT محاسبه کنید.

پروسس	زمان ورود به سیستم	زمان مورد نیاز پردازش
A	0	9
B	2	4
C	0	8
D	3	2
E	5	1

پاسخ: نمودار گانت به صورت زیر است:

C	B	D	E	B	C	A	
0	2	3	5	6	9	15	24

زمان انتظار هر یک از فرایندها برابر است با:

$$A = 24, B = 7, C = 15, D = 2, E = 1 \Rightarrow \frac{24 + 7 + 15 + 2 + 1}{5} = \frac{49}{5}$$

بالاترین نسبت پاسخ (HRRN)

در این روش برای هر فرایند نسبت پاسخی از رابطه $\frac{W+S}{S}$ ، بدست می‌آید، سپس فرایندی اجرا می‌شود که بالاترین نسبت پاسخ را دارد.

(W = زمان انتظار برای CPU) و (S = زمان اجرا) و ($W+S$ = زمان پاسخ (P))

ویژگی های الگوریتم HRRN

- ۱- گرسنگی ندارد.
- ۲- از معایب این روش، نیاز به تخمین زمان خدمت مورد نیاز قبل از به کارگیری می‌باشد.
- ۳- تابع انتخاب آن برابر $MAX(\frac{W+S}{S})$ است.
- ۴- سربار می‌تواند زیاد باشد.
- ۵- توان عملیاتی زیاد است.
- ۶- زمان بندی انحصاری (بدون قبضه کردن) است.

فیدبک (FB)

اگر نتوانیم روی زمان باقیمانده برای اجرا تمرکز کنیم، بهتر است روی زمان اجرای سپری شده تمرکز کنیم. زمان بندی FB براساس قبضه کردن صورت می‌گیرد و از یک روش اولویت پویا استفاده می‌شود. فرایندی که ابتدا وارد شود به صف 0 می‌رود و هنگامی که بعد از اولین اجرا به حالت آماده می‌رود در صف 1 با اولویت کمتر قرار می‌گیرد پس از هر اجرا به صف کم اولویت تر بعدی می‌رود. فرایند کوتاه بدون انتقال به صفهای پایین تر به سرعت اجرا می‌شود، اما فرایندهای طولانی به صفهای پایین می‌روند و فرایندهای جدیدتر و کوتاهتر به فرایندهای قدیمی تر و بلند تر ارجحیت دارند. در صف با کمترین اولویت از سیاست RR و در صفهای دیگر از سیاست FCFS استفاده می‌شود. (چون فرایند در صف کمترین اولویت، نمی‌تواند به صفهای پایین تر برود).

✍ زمان بندی FB غیر انحصاری می‌باشد، چون فرایندی از صفی به صف دیگر می‌تواند منتقل شود.

زمان بندی های زیر در کتاب سیلبرشاتس آورده شده است.

زمان بندی صف بازخوردی چند سطحی (MLFQ)

این زمانبند غیر انحصاری، به فرایندها اجازه می‌دهد تا از صفی به صف دیگر منتقل شوند. فرایندی که زیاد CPU را در اختیار داشته به صف کم اولویت‌تر می‌رود و فرایند I/O bound و محاوره‌ای به صف با اولویت بالاتر می‌رود و فرایندی که زیاد در صف با اولویت پایین بوده به صف با اولویت بالاتر می‌رود، یعنی کهنگی (سالمندی) از ایجاد گرسنگی جلوگیری می‌کند.

مثال

زمانبند صف بازخوردی چند سطحی (MLFQ) با 3 صف (0 تا 2) را در نظر بگیرید. صف اول از روش RR با کوانتوم 8 میلی ثانیه و صف دوم نیز از روش RR با کوانتوم 16 میلی ثانیه و صف آخر از روش FCFS استفاده می‌کند. زمانبند ابتدا تمامی فرایندهای موجود در صف 0 را اجرا می‌کند، اگر صف 0 خالی بود فرایندهای صف 1 اجرا می‌شود و اگر صف 0 و 1 خالی بود فرایندهای صف 2 اجرا می‌شود. فرایند تازه وارد در صف 0 قرار می‌گیرد اگر در مدت 8 میلی ثانیه اجرای آن به پایان نرسد به انتهای صف 1 می‌رود. به فرایند موجود در ابتدای صف 1، کوانتوم زمانی 16 میلی ثانیه داده می‌شود که اگر اجرای آن در این مدت تمام نشود به صف 2 می‌رود. اگر هر دو صف 0 و 1 خالی باشد، فرایندهای صف 2 براساس FCFS اجرا می‌شوند. بنابراین فرایندهای طولانی به صف 2 می‌روند و به ترتیب FCFS اجرا می‌شوند. ■

مثال

یک سیستم تک پردازنده ای با صف بازخورد چند سطحی (MLFQ) را در نظر بگیرید. به صف اول تکه زمانی 8 میکرو ثانیه، به صف دوم، تکه زمانی 16 میکرو ثانیه داده می‌شود. همچنین صف سوم با روش FCFS زمان بندی می‌شود. میانگین زمان پاسخ و میانگین زمان انتظار چقدر خواهد بود؟

فرایند	زمان اجرا
A	4
B	7
C	12
D	20
E	25
F	30

حل:

ابتدا برنامه‌ها وارد صف اول می‌شوند و در صورت نیاز به بیش از ۸ میکروثانیه به صف دوم وارد می‌شوند (فرایندهای F, E, D, C) و در صف دوم در صورت نیاز به بیش از ۱۶ میکروثانیه به صف سوم منتقل می‌شوند (فرایندهای F, E) و در صف سوم به روش FCFS به آنها رسیدگی می‌شود:

A	B	C	D	E	F	C	D	E	F	E	F	
0	4	11	19	27	35	43	47	59	75	91	92	98

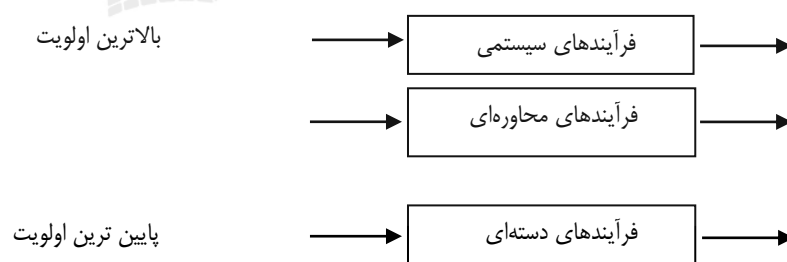
میانگین زمان انتظار برابر است با :

$$\frac{(4-4) + (11-7) + (47-12) + (59-20) + (92-25) + (98-30)}{6} = \frac{213}{6} = 35.5$$

زمان بندی صف چند سطحی (MLQ)

این الگوریتم صف آماده را به چند بخش تقسیم می‌کند. هر فرایند براساس صفات خود (اولویت، نوع) در صفی قرار می‌گیرد، که در هر صف الگوریتم زمان بندی خاصی وجود دارد. به طور معمول فرایندها به دو نوع ForeGround و BackGround تقسیم می‌شوند. اولویت فرایندهای پیش‌زمینه ممکن است از پس‌زمینه بالاتر باشد. در صف پیش‌زمینه از الگوریتم RR و در صف پس‌زمینه از الگوریتم FCFS استفاده می‌شود. همچنین بین صفها نیز باید زمان بندی وجود داشته باشد. که بر اساس الگوریتم غیرانحصاری و با اولویت ثابت پیاده‌سازی می‌شود.

شکل زیر الگوریتم زمان بندی صف چند سطحی با ۳ صف را نشان می‌دهد:



در این مثال، هیچ فرایندی در صف محاوره‌ای نمی‌تواند اجرا شود، مگر اینکه صف مربوط به فرایندهای سیستمی خالی باشد. همچنین اگر در حین اجرای فرایند دسته‌ای، یک فرایند محاوره‌ای وارد صف شود، فرایند دسته‌ای قبضه می‌شود، چون اولویت فرایند محاوره‌ای بالاتر است.

زمان بندی اولویت (Priority)

در این الگوریتم به هر فرایند اولییتی داده می‌شود و CPU به فرایندی داده می‌شود که بالاترین اولویت را دارد و فرایندهایی که اولویت آنها یکسان باشد به ترتیب FCFS زمان بندی می‌شوند.

مثال

میانگین زمان انتظار برای پنج فرایند زیر را در روش زمان بندی اولویت بدست آورید.
(اعداد کوچک، اولویت بالا را نشان می‌دهد.)

اولویت	زمان اجرا	فرایند
3	10	P1
1	1	P2
3	2	P3
4	1	P4
2	5	P5

حل: گانت فرایندها به صورت زیر است:

p_2	p_5	p_1	p_3	p_4	
0	1	6	16	18	19

میانگین زمان انتظار :

$$\frac{(16-10) + (1-1) + (18-2) + (19-1) + (6-5)}{4} = 8.2$$



مثال

برای سه پردازش زیر، زمان متوسط پاسخگویی در روش اولویت کدام است؟
(عدد کمتر در ستون اولویت، نشان دهنده اولویت بیشتر است.)

زمان اجرا	زمان ورود	اولویت	پردازش
4	t	1	P ₁
2	t	3	P ₂
1	t+3	2	P ₃

حل: ابتدا فرایند P1 اجرا می شود (چون اولویت آن بیشتر است) و سپس فرایند P3 و در نهایت فرایند P2 اجرا می شود. گانت به صورت زیر می باشد:

p_1	p_3	p_2	
0	$t+4$	$t+5$	$t+7$

بنابراین میانگین زمان پاسخگویی برابر است با :

$$= \frac{(t+4-t) + (t+7-t) + (t+5-t-3)}{3} = \frac{13}{3}$$



زمانبندی LCFS

در این روش، آخرین ورودی ابتدا سرویس می گیرد. این روش می تواند قبضه شدنی یا قبضه نشدنی باشد.

LCFS : Last-Come First-Served

مثال

پنج فرایند زیر به سیستمی وارد شده است. نمودار گانت را در صورت استفاده از الگوریتم LCFS انحصاری (NP-LCFS) را محاسبه کنید.

فرایند	زمان ورود	زمان پردازش
A	0	1.5
B	1	1.5
C	2	1.5
D	3	1.5
E	4	1.5

حل:

A	B	D	E	C	
0	1.5	3	4.5	6	7.5

پردازنده در لحظه 0 به تنها فرایند موجود یعنی A داده می شود و چون قبضه شدنی نمی باشد، تا پایان اجرای فرایند A ، پردازنده را در اختیار دارد. در زمان 1.5 فقط فرایند B رسیده و پردازنده را می گیرد. در زمان 3 که C و D حاضرند، ابتدا پردازنده به D داده می شود، چون آخرین ورودی است. بعد از اجرای D ، چون E نیز رسیده است، پردازنده به E داده می شود و در نهایت به C داده می شود.



مثال

با توجه به جدول زیر، نمودار گانت را رسم کنید. (برای فرایندها از الگوریتم Round-Robin با برش زمانی $q=2$ و برای نخ‌های درون هر فرآیند، از الگوریتم LCFS استفاده کنید).

فرایند	نخ	زمان پردازش	زمان ورود
	T11	1.5	0
	T12	1.5	1
	T21	2.5	2
	T22	2	3

حل: نمودار گانت به صورت زیر می باشد:

T_{11}	T_{12}	T_{21}	T_{22}	T_{12}	T_{11}	T_{22}	T_{21}	
0	1	2	3	4	4.5	5	6	7.5

متوسط زمان بازگشت نخ‌های فرایند P1 برابر است با:

$$P1: \frac{(5-0) + (4.5-1)}{2} = 4.25$$



زمانبندی در سیستم چند پردازنده ای

الگوریتم‌های متداول برای زمانبندی سیستم‌های چند پردازنده ای عبارتند از:

LPT - ۱ RPT - ۲ SPT - ۳

الگوریتم LPT

این الگوریتم از بین کارهای باقیمانده، طولانی‌ترین کار را برای اجرا انتخاب می‌کند. این الگوریتم بهینه نیست ولی معمولاً منجر به زمانبندی‌هایی با طول معقول می‌گردد.

مثال

طول زمانبندی را در سیستم تکالیف {8,4,1,7,2,13,2,6} در حالت دو پردازنده محاسبه کنید.

حل: ابتدا کارها را به ترتیب نزولی مرتب می‌کنیم:

$$\{\tau_i\} = \{13, 8, 7, 6, 4, 2, 2, 1\}$$

سپس کارها را به ترتیب به پردازنده‌ها داده و هر پردازنده ای که کارش را انجام داد، کار بعدی را اجرا خواهد کرد:

P1	13	6	2	1
P2	8	7	4	2

بنابراین طول زمانبندی برابر ۲۲ خواهد بود. (زمان مشغول بودن پردازنده اول که بیشترین زمان است).

مثال

طول زمانبندی را در سیستم تکالیف $\{8,4,1,7,2,13,2,6\}$ در حالت سه پردازنده محاسبه کنید.

حل: ابتدا کارها را به ترتیب نزولی مرتب می‌کنیم:

$$\{\tau_i\} = \{13,8,7,6,4,2,2,1\}$$

سپس کارها را به ترتیب به پردازنده‌ها داده و هر پردازنده‌ای که کارش را انجام داد، کار بعدی را اجرا خواهد کرد:

P1	13		2
P2	8	4	2
P3	7	6	1

بنابراین طول زمانبندی برابر ۱۵ خواهد بود. (زمان مشغول بودن پردازنده اول که بیشترین زمان است).

الگوریتم RPT

این الگوریتم از نظر طول زمانبندی مشابه LPT عمل کرده ولی زمان پاسخ بهتری را می‌دهد. در واقع RPT مانند LPT ای است که ترتیب تکالیف هر پردازنده معکوس شده است.

مثال

نتیجه استفاده از RPT برای $\{\tau_i\} = \{13,8,7,6,4,2,2,1\}$ در شرایط دو پردازنده را مشخص نمایید؟
حل: به روش LPT داریم:

P1	13	6	2	1
P2	8	7	4	2

که با معکوس کردن ترتیب تکالیف، خواهیم داشت

P1	1	2	6	13
P2	2	4	7	8

الگوریتم SPT

این الگوریتم چندپردازنده‌ای، ابتدا کارها را بر اساس زمان اجرای افزایشی مرتب کرده و سپس m تا کار اول را برای اجرا در m پردازنده موجود زمانبندی می‌کند (یک کار برای هر یک از پردازنده‌ها)، سپس m تا کار بعدی زمانبندی می‌گردند و به همین ترتیب تا آخر.

مثال

طول زمانبندی در سیستم تکالیف {8,4,1,7,2,13,2,6} با دو پردازنده به روش SPT را تعیین نمایید.

حل: ابتدا کارها را به ترتیب صعودی مرتب می کنیم:

{1,2,2,4,6,7,8,13}

سپس کارها را به ترتیب به پردازنده ها داده و هر پردازنده ای که کارش را انجام داد، کار بعدی را اجرا خواهد کرد:

P1	1	2	6	8
P2	2	4	7	13

بنابراین طول زمانبندی برابر ۲۶ خواهد بود. (زمان مشغول بودن پردازنده دوم که بیشترین زمان است).

■

مشخصات سیاست های زمان بندی

FB	HRRN	SRT	SPN	RR	FCFS	
	$\max(\frac{w+s}{s})$	Min[s-e]	Min[s]	ثابت	MAX[W]	تابع انتخاب
با قبضه کردن (در برهه زمانی)	بدون قبضه کردن	با قبضه کردن در ورود	بدون قبضه کردن	با قبضه کردن (در برهه زمانی)	بدون قبضه کردن	حالت تصمیم گیری
تاکید نشده است.	زیاد	زیاد	زیاد	اگر برهه زمانی خیلی کوچک باشد، کم می شود.	تاکید نشده است	توان عملیاتی
تاکید نشده است	زمان پاسخ خوبی را ارائه می کند.	زمان پاسخ خوبی را ارائه می کند.	برای فرایندهای کوتاه زمان پاسخ خوبی را ارائه می دهد.	برای فرایندهای کوتاه زمان پاسخ خوبی را ارائه می دهد.	می تواند زیاد باشد، به خصوص اگر واریانس زمانهای اجرا خیلی بزرگ باشد.	زمان پاسخ
می تواند زیاد باشد	می تواند زیاد باشد	می تواند زیاد باشد	می تواند زیاد باشد	کم	حداقل	سر بار
می تواند به نفع فرایندهای در I/O تنگنای باشد.	توازن مناسب	به فرایندهای طولانی صدمه می زند.	به فرایندهای طولانی صدمه می زند.	عملکرد عادلانه	به فرایندهای کوتاه و فرایندهای در تنگنای I/O صدمه می زند.	تاثیر بر روی فرایندها
امکان دارد	خیر	امکان دارد	امکان دارد	خیر	خیر	گرسنگی

w = زمان سپری شده در سیستم برای انتظار و اجرا تا به حال

e = زمان سپری شده، برای اجرا تا به حال

S = کل زمان مورد نیاز فرایند، که شامل e نیز هست.

شرط اینکه یک سیستم بلادرنگ قابل زمان بندی باشد این است که : $\sum_{i=1}^m \frac{c_i}{p_i} \leq 1$ 

c_i : زمان اجرای رخداد i p_i : تناوب رخداد i m : تعداد رخدادها