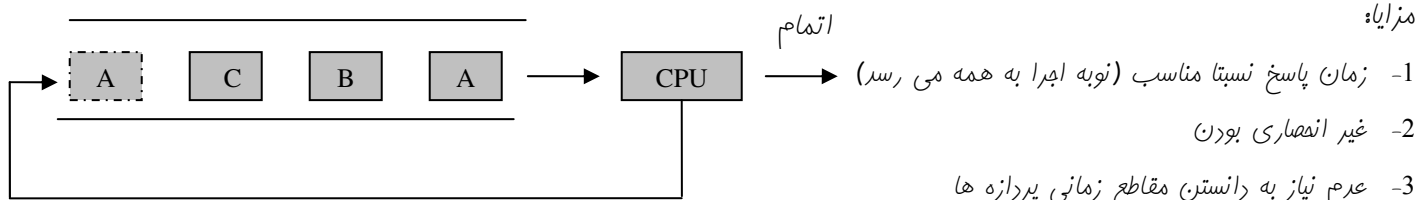


**الگوریتم نوبتی (Round Robin: RR) :** در این الگوریتم با در نظر گرفتن یک برهه زمانی (time slice) یا کوانتوم زمانی پردازشگرهای موجود در صف آماده هر کدام به اندازه این برهه زمانی CPU را بدست آورده به طوری که اگر در برهه زمانی شان اجرای خود را به اتمام نرسانند مجدداً به انتهای صف منتقل می شوند.

- این الگوریتم بیشتر در روش های اشتراک زمانی (Time sharing) استفاده می شود.

مزایا:



معایب:

- نیاز به دقت در تنظیم برش زمانی
  - کاهش کارایی پردازنده به خاطر زمان های تعویض متن در برهه های زمانی خیلی کوتاه
  - شباهت به الگوریتم FCFS برای برش های زمانی طولانی
- کوانتوم زمانی بین 10 تا 100 میلی ثانیه است .

**مثال:** سه پردازشگر زیر را در نظر بگیرید ، اگر از کوانتوم زمانی 4 میلی ثانیه استفاده شود ، میانگین زمان انتظار و میانگین زمان گردش کار را بیابید. (هر سه در لحظه صفر وارد شده اند ولی الویت به ترتیب با شماره است)

نمودار گانت پردازشگرها به صورت زیر خواهد بود.

پردازشگر	زمان اجرا
$p_1$	24
$p_2$	3
$p_3$	3

P1	P2	P3	P1	P1	P1	P1	P1	
0	4	7	10	14	18	22	26	30

$$\text{میانگین زمان انتظار} = \frac{6+4+7}{3} = \frac{17}{3} \text{ ms} \quad \text{میانگین زمان گردش کار} = \frac{(6+24) + (4+3) + (7+3)}{3} = \frac{47}{3} \text{ ms}$$

**مثال:**

چهار پردازنده مطابق جدول زیر در سیستم وجود دارند. اگر از روش RR با برش زمانی 1ms استفاده شود و از سربار ناشی از تعویض متن فرایندها صرف نظر شود میانگین زمان انتظار پردازشگرها چقدر است (فرض شود پردازشگرهای که از راه می رسد به ابتدای صف منتقل می شود)

پردازشگرها	زمان ورود	زمان پردازش
$p_1$	0	3
$p_2$	1	5
$p_3$	3	2
$p_4$	9	2

P1	P2	P1	P3	P2	P1	P3	P2	P2	P4	P2	P4	
0	1	2	3	4	5	6	7	8	9	10	11	12

$$\text{میانگین زمان انتظار} = \frac{3+5+2+1}{4} = \frac{11}{4} \text{ ms}$$

## الگوریتم زمانبندی الویت دار

الویت با بالاترین نسبت پاسخ (Highest Responses Ratio Next -HRN)

$$\text{الویت} = \frac{\text{زمان سرویس} + \text{زمان انتظار}}{\text{زمان سرویس}}$$

الگوریتم های الویت دار با در نظر گرفتن یک الویت برای هر کدام از پردازش های موجود در سیستم ، پردازش ای که بالاترین الویت را داشته باشد اجرایش را به صورت انحصاری انجام می دهد ، الویت هائی که می توان در این الگوریتم در نظر گرفت عبارتند از

1- نسبت مقاطع زمانی ورودی یا خروجی به مقاطع زمانی پردازنده (CPU Burst به I/O Burst)

2- محدودیت زمانی ، نیاز های حافظه

3- تعداد فایل های باز شده

4- اهمیت پردازش و بخش ارائه دهنده کار

5- الویت نسبت پاسخ (فرمول بالائی)

مزایا:

1- طبق این الگوریتم پردازش های کوتاه در ابتدا الویت بالا پیدا می کنند ، چون زمان سرویس (زمان مورد نیاز جهت اجرا) کمتری نیاز دارند

2- پردازش هائی که مدت زیادی منتظر مانده اند نیز الویت بالا پیدا می کنند ، پس این شانس را پیدا می کنند که اجرا شوند.

معایب:

1- می بایست زمان سرویس را از قبل معلوم کرد

2- انحصاری است

مثال:

پردازش های زیر را در نظر بگیرید که همگی در لحظه ی صفر به ترتیب داده شده رسیده اند میانگین زمان انتظار را بیابید و جهت زمانبندی پردازش ها از الگوریتم الویت دار استفاده کنید و پردازش ای که عدد الویت آن کمتر است ، الویت بالاتری دارد.

حل: نمودار گانت به صورت زیر خواهد بود

الویت	زمان اجرا	پردازش ها
3	10	$p_1$
1	1	$p_2$
3	2	$p_3$
4	1	$p_4$
2	5	$p_5$

P2	P5	P1	P3	P4
0	1	6	16	18
				19

$$\text{میانگین زمان انتظار} = \frac{6+0+16+18+1}{5} = \frac{41}{5} \text{ ms}$$

## الگوریتم صف چند سطحی (Multi Level Queue: MLQ):

در الگوریتم صف چند سطحی پردازنده ها در صف های مختلف ، که هر صف الویت خاصی دارد قرار می گیرند و در صف های مختلف از الگوریتم های زمانبندی مختلفی استفاده می شود ، پارامتر هائی که در این الگوریتم می بایست مشخص نمود عبارتند از :

1- تعداد صف ها

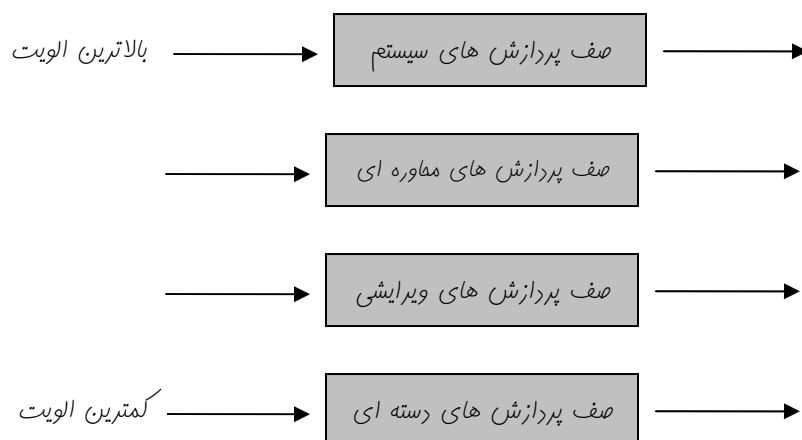
2- الگوریتم زمانبندی استفاده شده در هر صف

3- الویت صف ها نسبت به هم

• اگر در صفی با الویت بالا پردازنده ای وجود داشته باشد CPU در ابتدا پردازنده های آن صف را سرویس داده و در صورتی که صف های الویت بالاتر خالی شود به سراغ صف الویت پایین می رود.

• به عنوان مثال سیستمی می تواند 4

صف آماده با الویت های زیر داشته باشد



## الگوریتم صف باز خورد چند سطحی (Multi Level Feedback Queue : MFQ)

این الگوریتم مانند الگوریتم MLQ می باشد با این تفاوت که در این الگوریتم امکان حرکت پردازنده ها بین صف های مختلف نیز وجود دارد ، در این الگوریتم علاوه بر مشخص نمودن پارامتر های MLQ موارد زیر نیز می بایست مشخص شود .

1- چه موقع یک پردازنده از صف بالا به صف پایین مهاجرت می کند

2- چه موقع یک پردازنده از صف پایین به صف بالا مهاجرت می کند

مثال:

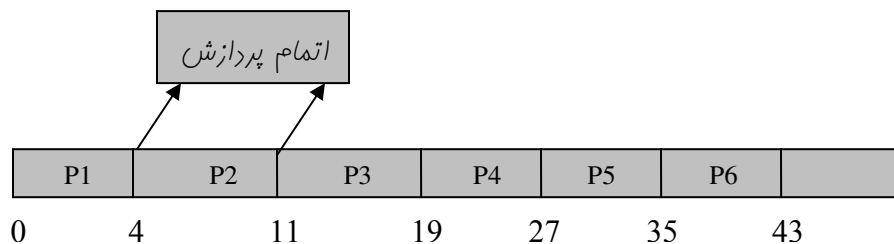
یک سیستم با الگوریتم MFQ یا MLF سه سطحی را در نظر بگیرید . اگر سطح اول از الگوریتم RR با کوانتوم زمانی 8 میکرو ثانیه و صف دوم از الگوریتم RR با کوانتوم زمانی 16 میکرو ثانیه و صف سوم با الگوریتم FCFS زمانبندی شوند . اگر 6 پردازنده همگی در زمان صفر با زمانهای اجرای 4 ، 7 ، 12 ، 20 ، 25 و 30 میکرو ثانیه وارد سیستم می شوند میانگین زمان برگشت و میانگین زمان انتظار

را در این سیستم بیابید (هر پردازش به معض پایان کوانتوم زمانی از صف بالاتر به صف پایین تر مهاجرت می کند) (کارشناسی ارشد - دولتی 80)

حل:

ابتدا برنامه ها در صف اول با  $TS=8 \mu S$  قرار می گیرند و اگر در مدت 8 میکرو ثانیه تمام نشوند به صف دوم با  $TS=16 \mu S$  منتقل می گردند باز هم اگر در این مدت 16 میکرو ثانیه تمام نشوند وارد صف سوم با الگوریتم FCFS می گردند. بنابراین نمودار زمانی این پردازش ها به صورت شکل زیر خواهد بود

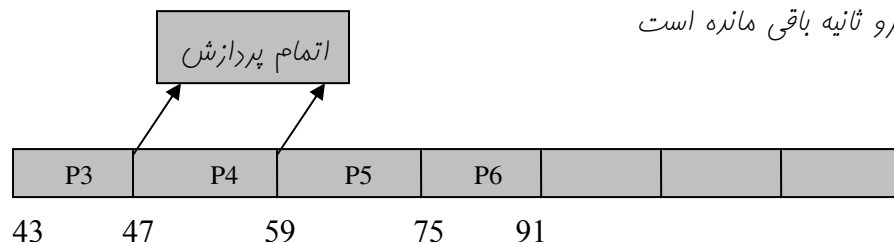
صف اول با  $TS=8 \mu S$



پس از یکبار اجرای پرفشی P1 و P2 تمام می شوند در حالی که از برنامه های P3 ، P4 ، P5 و P6

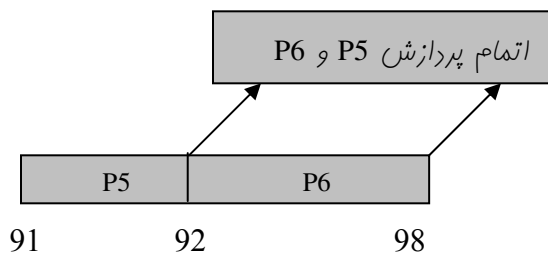
به ترتیب زمان های 4 ، 12 ، 17 و 22 میکرو ثانیه باقی مانده است

صف دوم با  $TS=16 \mu s$



در انتهای مرحله ی دوم P3 و P4 تمام شده و از برنامه P5 به اندازه 1 و از برنامه P6 به اندازه 6 میکرو ثانیه باقی مانده است ، این دو پردازش در مرحله ی آخر به صورت FCFS زمانبندی می شوند.

صف سوم FCFS.



$$\text{میانگین زمان انتظار} = \frac{0 + 4 + 35 + 39 + 67 + 68}{6} = 35/5$$

$$\text{میانگین زمان برگشت} = \frac{(0 + 4) + (4 + 7) + (35 + 12) + (39 + 20) + (67 + 25) + (68 + 30)}{6} = 51/83$$

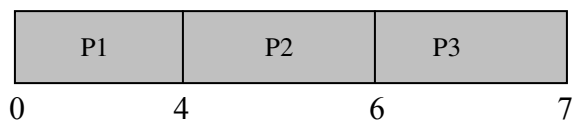
$p_3$	1	$t+3$	1
-------	---	-------	---

مثال: سه پردازش دسته ای ( $P_1$ ،  $P_2$ ،  $P_3$ ) با زمان اجرا و زمان وارد شدن زیر را در نظر بگیرید متوسط زمان پاسکوئی را با تمامی الگوریتم ها به جز الگوریتم های MFQ و MLQ بیابید. (ارشد-83)

	الویت	زمان وارد شدن	زمان اجرا
$p_1$	2	$t$	4
$p_2$	0	$t$	2

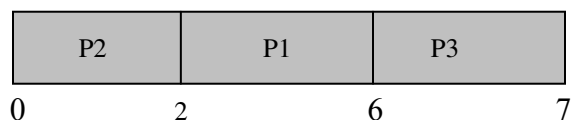
حل: t, 1 صفر در نظر می گیریم

الف) FCFS



$$AWT = \frac{(0+4) + (4+2) + (3+1)}{3} = \frac{14}{3}$$

ب) SJF



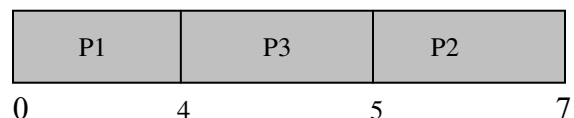
$$AWT = \frac{(2+4) + (0+2) + (3+1)}{3} = \frac{12}{3}$$

ج) SRT



$$AWT = \frac{(3+4) + (0+2) + (0+1)}{3} = \frac{10}{3}$$

د) الگوریتم نوبتی (Priority):



$$AWT = \frac{(0+4) + (5+2) + (1+1)}{3} = \frac{13}{3}$$

مثال:

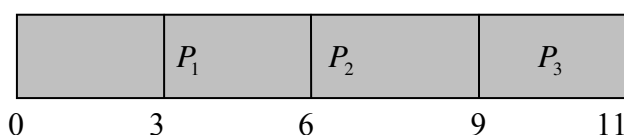
چهار پردازش  $P_0$ ،  $P_1$ ،  $P_2$ ،  $P_3$  با مشخصات زیر در نظر بگیرید. میانگین زمان پاسخ دهی آنها را در الگوریتم های زمانبندی زیر بیابید. (فرض کنید تکه زمانی معادل یک واحد زمانی است. در مورد سیاست باگرددش نوبت پردازش ای که وارد سیستم می شود در همان ابتدای ورودش اجرای آن آغاز می شود)

نام پردازش	زمان ورود به سیستم	زمان پردازش
$P_0$	0	3
$P_1$	1	3
$P_2$	4	3
$P_3$	6	2

الف) FCFS

ب) RR

حل: الف) FCFS



$$AWT = \frac{(0+3) + (2+3) + (2+3) + (3+2)}{4} = \frac{18}{4}$$

ب) RR

$P_0$	$P_1$	$P_0$	$P_1$	$P_2$	$P_0$	$P_3$	$P_1$	$P_2$	$P_3$	$P_2$
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

$$0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad 9$$

$$AWT = \frac{(3+10) + (4+3) + (4+3) + (2+2)}{4} = \frac{24}{4}$$

**Pnu-Soal.ir**