

تمرین پنجم سیستم‌های عامل

اشکان شکيبا (۹۹۳۱۰۳۰)

سوال اول

R1: منبع چپ، R2: منبع راست، R3: منبع پایین

جدول تخصیص منابع را بر اساس گراف آن تکمیل می‌کنیم.

	allocated			available			need			
	R1	R2	R3	R1	R2	R3	R1	R2	R3	Finished
P0	1	0	4	0	0	1	0	1	0	T
P1	1	1	0	0	4	1	1	0	0	T
P2	0	1	0	4	4	2	0	0	1	T
P3	0	1	0	2	2	2	0	0	2	T

P0: 0 0 1 < 0 1 0

P1: 0 0 1 < 1 0 0

P2: 0 0 1 >= 0 0 1 \Rightarrow new value = 0 0 1 + 0 1 0 = 0 1 1

P3: 0 1 1 < 0 0 2

P0: 0 1 1 >= 0 1 0 \Rightarrow new value = 0 1 1 + 1 0 1 = 1 1 2

P1: 1 1 2 >= 1 0 0 \Rightarrow new value = 1 1 2 + 1 1 0 = 2 2 2

P3: 2 2 2 >= 0 0 2 \Rightarrow new value = 2 2 2 + 0 1 0 = 2 3 2

با توجه به اینکه در انتها، به مقدار $2^3 2$ رسیدیم که برابر با منابع موجود در گراف است، پس می‌توان نتیجه گرفت بن‌بستی وجود ندارد و می‌توانیم فرایندها را با ترتیب P_2, P_0, P_1, P_3 اجرا کنیم.

سوال دوم

بعد از تکمیل ستون need، از الگوریتم بانکدار استفاده می‌کنیم.

	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
P0	2	1	0	6	1	0	0	4	1	1	2	3	1	1	0	2
P1	0	5	7	1	0	0	1	1	2	1	2	7	0	5	6	0
P2	6	5	2	3	4	5	2	1	6	6	4	8	2	0	0	2
P3	3	5	6	1	3	3	6	0	9	9	10	8	0	2	0	1
P4	6	5	6	1	2	1	2	0	11	10	12	8	4	4	4	1

$$P_0: 1\ 1\ 2\ 3 \geq 1\ 1\ 0\ 2 \Rightarrow \text{new value} = 1\ 1\ 2\ 3 + 1\ 0\ 0\ 4 = 2\ 1\ 2\ 7$$

$$P_1: 2\ 1\ 2\ 7 < 0\ 5\ 6\ 0$$

$$P_2: 2\ 1\ 2\ 7 \geq 2\ 0\ 0\ 2 \Rightarrow \text{new value} = 2\ 1\ 2\ 7 + 4\ 5\ 2\ 1 = 6\ 6\ 4\ 8$$

$$P_3: 6\ 6\ 4\ 8 \geq 0\ 2\ 0\ 1 \Rightarrow \text{new value} = 6\ 6\ 4\ 8 + 3\ 3\ 6\ 0 = 9\ 9\ 10\ 8$$

$$P_4: 9\ 9\ 10\ 8 \geq 4\ 4\ 4\ 1 \Rightarrow \text{new value} = 9\ 9\ 10\ 8 + 2\ 1\ 2\ 0 = 11\ 10\ 12\ 8$$

$$P_1: 11\ 10\ 12\ 8 \geq 0\ 5\ 6\ 0 \Rightarrow \text{new value} = 11\ 10\ 12\ 8 + 0\ 0\ 1\ 1 = 11\ 10\ 13\ 9$$

بنابراین سیستم در حالت امن است و می‌توان پردازنده‌ها را با ترتیب P_0, P_2, P_3, P_4, P_1 اجرا کرد.

سوال سوم

الف) حداقل هفت R.

اگر شش تا داشته باشیم و همه یک R کمتر از نیازشان در اختیار داشته باشند، می‌تواند منجر به بن‌بست شود؛ اما با وجود هفت R در هیچ حالتی امکان وقوع بن‌بست نیست.

ب) حداکثر سه R.

اگر چهارتا داشته باشیم، می‌توان منابع را طوری مدیریت کرد که بن‌بست رخ ندهد.

سوال چهارم

الف) با فرض نبود TLB، برای دسترسی به هر داده نیاز به دو بار مراجعه به حافظه اصلی خواهیم داشت؛ یک بار مراجعه به page table و یک بار برای دسترسی به داده مورد نظر. بنابراین $100 \text{ ns} = 50 * 2$ زمان می‌برد.

ب)

$$\begin{aligned} \text{زمان موثر دسترسی} &= \text{TLB hit rate} (\text{TLB access time} + \text{memory access time}) + (1 - \text{TLB hit rate}) (\text{TLB access time} + 2 * \text{memory access time}) \\ &= 0.75 (2 + 50) + 0.25 (2 + 100) = 64.5 \text{ ns} \end{aligned}$$

سوال پنجم

حافظه مورد نیاز:

$$83/100 * 400 = 332 \text{ MB}$$

با توجه به اینکه در سیاست اولین مناسب، برای تخصیص هر تعداد بلاک، نصف همان تعداد به دلیل fragmentation هدر می‌رود، پس می‌توان نوشت:

$$\text{پاسخ} = 332 * 3/2 = 498 \text{ MB}$$

سوال ششم

(الف)

FIFO

3	9	2	1	5	3	2	4	9	1	0	0	1	5	1	2	9
3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4
	9	9	9	9	9	9	9	9	9	0	0	0	0	0	0	0
		2	2	2	2	2	2	2	2	2	2	2	2	2	2	9
			1	1	1	1	1	1	1	1	1	1	1	1	1	1
				5	5	5	5	5	5	5	5	5	5	5	5	5
F	F	F	F	F			F			F						F

۸ خطا

4	7	7	0	4	0	7	3	3	1	9	7	4	0	3	4	7
4	4	4	4	4	4	4	4	4	4	9	9	9	9	9	9	9
	7	7	7	7	7	7	7	7	7	7	7	4	4	4	4	4
			0	0	0	0	0	0	0	0	0	0	0	0	0	7
							3	3	3	3	3	3	3	3	3	3
									1	1	1	1	1	1	1	1
F	F		F				F		F	F		F				F

٨ خطا

LRU

3	9	2	1	5	3	2	4	9	1	0	0	1	5	1	2	9
3	3	3	3	3	3	3	3	3	3	0	0	0	0	0	0	0
	9	9	9	9	9	9	4	4	4	4	4	4	4	4	2	2
		2	2	2	2	2	2	2	2	2	2	2	5	5	5	5
			1	1	1	1	1	9	9	9	9	9	9	9	9	9
				5	5	5	5	5	1	1	1	1	1	1	1	1
F	F	F	F	F			F	F	F	F			F		F	

١١ خطا

4	7	7	0	4	0	7	3	3	1	9	7	4	0	3	4	7
4	4	4	4	4	4	4	4	4	4	9	9	9	9	9	9	9
	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
			0	0	0	0	0	0	0	0	0	4	4	4	4	4
							3	3	3	3	3	3	0	0	0	0
									1	1	1	1	1	3	3	3
F	F		F				F		F	F		F	F	F		

٩ خطا

Optimal

3	9	2	1	5	3	2	4	9	1	0	0	1	5	1	2	9
3	3	3	3	3	3	3	4	4	4	0	0	0	0	0	0	0
	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
			1	1	1	1	1	1	1	1	1	1	1	1	1	1
				5	5	5	5	5	5	5	5	5	5	5	5	5
F	F	F	F	F			F			F						

٧ خطا

4	7	7	0	4	0	7	3	3	1	9	7	4	0	3	4	7
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
			0	0	0	0	0	0	0	0	0	0	0	0	0	0
							3	3	3	3	3	3	3	3	3	3
									1	9	9	9	9	9	9	9
F	F		F				F		F	F						

۶ خطا

ب) با فرض اینکه داده‌ها در همه صد دسترسی اول متفاوت باشند، در همه آنها page fault خواهیم داشت. حال اگر به همین داده‌ها به صورت وارونه دسترسی پیدا کنیم، ۴ صفحه آن از پیش در جدول حضور دارند و برای بقیه دسترسی‌ها با page fault مواجه خواهیم شد.

بنابراین در کل $196 = (100 - 4) + 100$ تا page fault خواهیم داشت.