



بسمه تعالی
سیستم های عامل
نیمسال اول ۹۹-۹۸
تمرین (۱۰)



دانشکده مهندسی کامپیوتر

مهلت تحویل: ۱۳۹۸/۰۹/۲۱

دانشگاه صنعتی امیرکبیر

شماره دانشجویی: ۹۶۳۱۰۰۱

نام و نام خانوادگی: محمدرضا اخگری

۱.

یکی از راه های جلوگیری از بن بست prevention بود. که یکی از روش های پیاده سازی این موضوع برطرف کردن mutual exclusion بود. این کار برای فایل های read-only قابل اجراست.

پس تعداد بن بست ها + میباشد. میشود فایل را به همه اختصاص داد.

۲.

اگر فرآیندها به صورتی تصور شوند که چند فرایند باید دقیقا موازی اجرا بشن (نه شبه موازی) مثلا فرایند های ۴ و ۵ و ۶ در واقع یکی از این فرایندها نمی تواند منتظر بماند تا دوفرایند دیگر تمام و منابع آنها آزاد شود و بعد خودش اجرا شود (بستگی به ماهیت فرایند ها دارد) در این حالت حداقل ۹ تا A نیاز است همین مورد برای فرایندهای ۲ و ۳ نیاز به حداقل ۵ تا B دارد .

در صورتی که فرض شود میتوان همروند اجرا کرد برای حالت های زیر مقدار AB نوشته شده است.

A-B	منابع مورد نیاز	اجرا در هر لحظه به صورت همروند
(4,2)	(4,2)	(P1)
(4,4)	(2,3) (3,2)	(P2,P3)
(2,3)	(2,3) (2,1)	(P2,P6)
(5,2)	(3,2) (4,1)	(P3,P4)
(7,2)	(4,1) (3,2) (2,1)	(P4,P5,P6)

بدین طریق میتوان فهمید به ۷ عدد A و ۴ عدد B نیاز دارد.

۳.

فرض کنید نیاز هر پردازنده به صورت s_i نمایش داده شود.



بسمه تعالی
سیستم های عامل
نیمسال اول ۹۸-۹۹
تمرین (۱۰)



دانشکده مهندسی کامپیوتر

مهلت تحویل: ۱۳۹۸/۰۹/۲۱

دانشگاه صنعتی امیرکبیر

شماره دانشجویی: ۹۶۳۱۰۰۱

نام و نام خانوادگی: محمدرضا اخگری

میدانیم طبق فرض $s_i \leq m$ است.

حال برای اینکه بن بست ایجاد نشود. در ابتدا به هر پردازنده یکی کمتر از مقدار نیازش اختصاص داده میشود.

حال کفایت یک منبع اختصاص داده شود. که مقدار این عبارت باید از m کمتر باشد.

$$\sum_{i=0}^n (s_i - 1) + 1 = M \rightarrow \sum_{i=0}^n s_i - n + 1 = m \rightarrow \sum_{i=0}^n s_i + 1 = n + m \rightarrow \sum_{i=0}^n s_i < n + m$$

با اختصاص یک منبع به یکی از فرآیندها، اجرا میشود و منابع خود را آزاد میکند.

۴.

قبل از اجابت درخواست:

Need Table	A	B	C
P0	3	5	6
P1	5	3	3
P2	2	1	3
P3	3	5	7
P4	1	2	3

$$avail = (1,2,3) \xrightarrow{P_4} avail = (3,5,6) \xrightarrow{P_0} avail = (6,11,14) \rightarrow \dots$$

پس ترتیبی پیدا میشود و امن است. (برای مثال $\langle P4, P0, P1, P2, P3 \rangle$)

بعد از اجابت درخواست:

در صورت اجابت درخواست دیگر از منبع اول نخواهیم داشت. و نمیتوان ترتیبی برای اجرا یافت (همه به A نیاز دارند) پس نا امن خواهد شد.