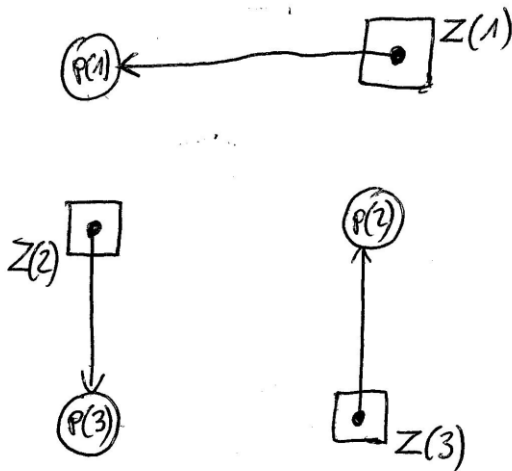


1. (2p.) Narysuj przykładowy graf przydziału (jednokrotnych) zasobów (bez zakleszczenia) i sprawdź, jakie przykładowe żądania przydzielenia zasobów spowodują powstanie zakleszczenia, a jakie nie.

1. Przykładowy graf przydziału jednokrotnych zasobów (bez zakleszczenia)



W przypadku równej ilości procesów i pojedynczych zasobów zakleszczenie zawsze nastąpi po żądaniu innego zasobu przez wszystkie procesy jednocześnie. Natomiast żądanie przez jeden z procesów dowolnego z zajętych zasobów nigdy nie doprowadzi do zakleszczenia. W zilustrowanym przypadku jeśli proces 3 zażąda zasobu numer 3 a proces 2 zażąda zasobu 1 - nie nastąpi zakleszczenie. Uogólniając, stworzenie powiązań generujących dowolny cykl w grafie przydziału jednokrotnych zasobów automatycznie powoduje zakleszczenie. Jeśli zasobów byłoby więcej niż procesów, ryzyko zakleszczenia byłoby mniejsze (z powodu większego wyboru zasobów i większej szansy na trafienie niezajętego) choć również możliwe, ponieważ niezależnie od ilości wolnych zasobów dany proces może żądać zasobu zajmowanego przez inny proces który z kolei potrzebuje zasobu zajmowanego przez ten pierwszy. Podobnie w przypadku większej liczby procesów niż zasobów - proces bez zasobów nie może stworzyć zakleszczenia, ponieważ nie zajmuje żadnego zasobu.

2. (5p.) W systemie są następujące liczby egzemplarzy zasobów:

- A: 10,
- B: 3,
- C: 1,
- D: 4,
- E: 5.

Aktualnie przydzielone zasoby (i maksymalne zapotrzebowania) są następujące:

Aktualny przydział						Maksymalne zapotrzebowanie					
	A	B	C	D	E		A	B	C	D	E
P1	2	0	0	1	0	P1	5	0	0	2	3
P2	3	2	0	2	0	P2	3	2	1	3	0
P3	0	0	1	0	3	P3	0	1	1	0	3
P4	3	0	0	0	0	P4	6	0	0	2	4
P5	0	0	0	0	2	P5	2	3	0	4	2

Czy ten system jest w stanie bezpiecznym? Podaj sekwencję kończenia procesów świadcząca o tym.

Korzystając z algorytmu bankiera sprawdzam czy system znajduje się w stanie bezpiecznym. Tablice PRZYDZIAŁ podano w treści zadania (na potrzeby indeksowania przyjmuję A=1, B=2...). Tablice DOSTĘPNE i POTRZEBY tworzone są na podstawie danych z treści zadania:

DOSTĘPNE = [2,1,0,1,0]

POTRZEBY					
	A	B	C	D	E
P1	3	0	0	1	3
P2	0	0	1	1	0
P3	0	1	0	0	0
P4	3	0	0	2	4
P5	2	3	0	4	0

Lista kroków:

1. KONIEC = [false, false, false, false, false]

ROBOCZE = DOSTĘPNE = [2,1,0,1,0]

2. dla $j = 3$ zachodzi $\text{Koniec}[j] = \text{true}$ i zarazem $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[3] = [2,1,1,1,3]$

$\text{Koniec}[3] = \text{true}$

3. Wracam do kroku 2: dla $j=2$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[2] = [5,3,1,3,3]$

$\text{Koniec}[2] = \text{true}$

4. Wracam do kroku 2: dla $j=1$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[1] = [7,3,1,4,3]$

$\text{Koniec}[1] = \text{true}$

5. Wracam do kroku 2: dla $j=5$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[5] = [7,3,1,4,5]$

$\text{Koniec}[5] = \text{true}$

6. Wracam do kroku 2: dla $j=4$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[4] = [10,3,1,4,5]$

$\text{Koniec}[4] = \text{true}$

Dla każdego $j = 1, 2, \dots, n$ $\text{Koniec}[j] = \text{true}$. Stan przedstawiony w treści zadania jest bezpieczny, a sekwencja ciągu bezpiecznego to (3,2,1,5,4)

3. (3p.) Co się stanie, gdy zgłoszone zostaną następujące żądania przydzielenia zasobów:

- P4: 2xD,
- P4: 2xA,
- P5: 1xD,

Jeżeli zasoby zostaną przydzielone, podaj odpowiednią sekwencję kończenia procesów. Jeżeli nie, podaj przyczynę.

Zgodnie z zasadą unikania zakleszczeń, przed każdym dodatkowym przydzieleniem zasobów system musi sprawdzić, czy spełnienie żądania doprowadziłoby do zakleszczenia. Jednakże w przypadku istniejącego skonstruowanego ciągu bezpiecznego, nie ma znaczenia, czy dany proces skorzysta ze swojego maksymalnego zapotrzebowania czy też nie, ponieważ bezpieczna sekwencja umożliwia zawsze wyjście z ciągu procesów niezależnie od ich realnego wykorzystania zasobów (względem maksymalnego deklarowanego).

Jeśli jednak pytanie dotyczy zwiększenia aktualnych przydziałów przed sprawdzeniem, czy da się wyjść z ciągu, wymaga to przeprowadzenia ponownie algorytmu bankiera dla zmodyfikowanych danych wejściowych (czyli tablicy PRZYDZIAŁ).

Jeśli tak:

	Aktualny przydział				
	A	B	C	D	E
P1	2	0	0	1	0
P2	3	2	0	2	0
P3	0	0	1	0	3
P4	5	0	0	2	0
P5	0	0	0	1	2

DOSTĘPNE = [0,1,0,-2,0] - odmówiony zostałby najprawdopodobniej (jeśli kolejność odpowiedzi na zapytania z pytania jest chronologiczna) przydział 1xD dla P5 oraz P4 dostałoby tylko jeden egzemplarz D, ponieważ zabrakłoby dwóch egzemplarzy zasobu D.

W tym przypadku rzeczywista tablica przydziału to:

	Aktualny przydział				
	A	B	C	D	E
P1	2	0	0	1	0
P2	3	2	0	2	0
P3	0	0	1	0	3
P4	5	0	0	1	0
P5	0	0	0	0	2

DOSTĘPNE = [0,1,0,0,0]

	POTRZEBY				
	A	B	C	D	E
P1	3	0	0	1	3
P2	0	0	1	1	0
P3	0	1	0	0	0
P4	1	0	0	1	4
P5	2	3	0	4	0

Lista kroków:

1. KONIEC = [false,false,false,false,false]

ROBOCZE = DOSTĘPNE = [0,1,0,0,0]

2. dla $j = 3$ zachodzi $\text{Koniec}[j] = \text{true}$ i zarazem $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[3] = [0,1,1,0,3]$

$\text{Koniec}[3] = \text{true}$

Nie można znaleźć innego j , dla którego potrzeby procesu byłyby \geq od zasobów roboczych. W związku z tym nie istniałby ciąg bezpieczny, byłaby szansa zakleszczenia jako że stan jest niebezpieczny. Żądanie $1 \times D$ mogłoby być na tej podstawie oddalone.

Wariant dla przydzielenia $1 \times D$ dla P5 zamiast P4:

	Aktualny przydział				
	A	B	C	D	E
P1	2	0	0	1	0
P2	3	2	0	2	0
P3	0	0	1	0	3
P4	5	0	0	0	0
P5	0	0	0	1	2

DOSTĘPNE = [0,1,0,0,0]

	POTRZEBY				
	A	B	C	D	E
P1	3	0	0	1	3
P2	0	0	1	1	0
P3	0	1	0	0	0
P4	1	0	0	2	4
P5	2	3	0	3	0

Lista kroków:

1. KONIEC = [false,false,false,false,false]

ROBOCZE = DOSTĘPNE = [0,1,0,0,0]

2. dla $j = 3$ zachodzi $\text{Koniec}[j] = \text{true}$ i zarazem $\text{Potrzeby}[j] \leq \text{Robocze}$

$\text{Robocze} = \text{Robocze} + \text{Przydział}[3] = [0,1,1,0,3]$

$\text{Koniec}[3] = \text{true}$

Nie można znaleźć innego j , dla którego potrzeby procesu byłyby \geq od zasobów roboczych. W związku z tym nie istniałby ciąg bezpieczny, byłaby szansa zakleszczenia jako że stan jest niebezpieczny. Żądanie $1 \times D$ mogłoby być na tej podstawie oddalone.

Wariant dla odmówienia przydziału zasobu D i przydzielenia tylko $2 \times A$ dla P4:

	Aktualny przydział				
	A	B	C	D	E
P1	2	0	0	1	0
P2	3	2	0	2	0
P3	0	0	1	0	3
P4	5	0	0	0	0

P5 0 0 0 0 2

DOSTĘPNE = [0,1,0,1,0]

POTRZEBY

	A	B	C	D	E
P1	3	0	0	1	3
P2	0	0	1	1	0
P3	0	1	0	0	0
P4	1	0	0	2	4
P5	2	3	0	4	0

Lista kroków:

1. KONIEC = [false,false,false,false,false]

ROBOCZE = DOSTĘPNE = [0,1,0,1,0]

2. dla $j = 3$ zachodzi $\text{Koniec}[j] = \text{true}$ i zarazem $\text{Potrzeby}[j] \leq \text{Robocze}$
 $\text{Robocze} = \text{Robocze} + \text{Przydział}[3] = [0,1,1,1,3]$
 $\text{Koniec}[3] = \text{true}$

3. Wracam do kroku 2: dla $j=2$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$
 $\text{Robocze} = \text{Robocze} + \text{Przydział}[2] = [3,3,1,3,3]$
 $\text{Koniec}[2] = \text{true}$

4. Wracam do kroku 2: dla $j=1$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$
 $\text{Robocze} = \text{Robocze} + \text{Przydział}[1] = [5,3,1,4,3]$
 $\text{Koniec}[1] = \text{true}$

5. Wracam do kroku 2: dla $j=5$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$
 $\text{Robocze} = \text{Robocze} + \text{Przydział}[5] = [5,3,1,4,5]$
 $\text{Koniec}[5] = \text{true}$

6. Wracam do kroku 2: dla $j=4$ $\text{Koniec}[j] = \text{true}$ i $\text{Potrzeby}[j] \leq \text{Robocze}$
 $\text{Robocze} = \text{Robocze} + \text{Przydział}[4] = [10,3,1,4,5]$
 $\text{Koniec}[4] = \text{true}$

Dla każdego $j = 1, 2, \dots, n$ $\text{Koniec}[j] = \text{true}$. Stan przedstawiony w treści zadania jest bezpieczny, a sekwencja ciągu bezpiecznego to (3,2,1,5,4).

Żądanie $2x_A$ dla P_4 zostanie spełnione, sekwencja bezpiecznego wykonania procesów nie zmieni się. Na żadnym pośrednim etapie (po wykonaniu któregoś z procesów) żądanie przez którykolwiek proces zasobu D doprowadziłoby do przejścia w stan niebezpieczny i zakleszczenia.