



Statki wpływające do portu

Zad 1. Do portu zawijają i odpływają z niego statki. Każdy z N statków po wpłynięciu do portu musi zająć jakieś wolne miejsce przy nabrzeżu. Liczba miejsc jest ograniczona i wynosi m : ($m < n$). Ponadto, w celu wejścia do portu lub wyjścia z portu statek i -ty potrzebuje k_i holowników. Łączną liczbą holowników wynosi h : ($h < \sum_{i=1}^n k_i$)

Napisać program dla procesu–statku zapewniający, że nie nastąpi zakleszczenie ani zagłodzenie któregoś ze statków.

Miejsca przy nabrzeżu i holowniki należy traktować jako zasoby.



Zad 2. Przy nabrzeżu stoi statek o pojemności N . Statek z ladem jest połączony mostkiem o pojemności K : ($K < N$).

Na statek próbują dostać się pasażerowie, z tym, że na statek nie może ich wejść więcej niż N , a wchodzić na statek na mostku nie może być ich równocześnie więcej niż K .

Statek co jedna godzinę wypływa w rejs. W momencie odpływania kapitan statku musi dopilnować aby na mostku nie było żadnego wchodzącego pasażera. Jednocześnie musi dopilnować by liczba pasażerów na statku nie przekroczyła N .

Napisz odpowiednio procedury Pasażer i Kapitan zsynchronizowane za pomocą jakiegokolwiek z metod synchronizacji procesów.



Implementacja semafora

Zad 2. Udowodnij, że poniższe implementacje operacji P i V w ogólnym semaforze s za pomocą semaforów binarnych $mutex_s$ i $delay_s$ oraz operacji P_b i V_b są nieprawidłowe.

P(s):

```
P_b(mutex_s);  
s := s - 1;  
V_b(mutex_s);  
if s < 0 then  
    P_b(delay_s);
```

V(s):

```
P_b(mutex_s);  
s := s + 1;  
V_b(mutex_s);  
if s <= 0 then  
    V_b(delay_s);
```



Zad 4. Wskaż czy w poniżej zaprezentowanym rozwiązaniu problemu czytelników i pisarzy spełnione są ogólne założenia tego problemu. Przeanalizuj czy rozwiązanie to prowadzi do zagłodzenia czytelników lub pisarzy.

```
procedure reader;  
begin  
  region v do rc := rc + 1;  
  read; //operacja poza regionem krytycznym  
  region v do rc := rc - 1;  
end  
  
procedure writer;  
begin  
  region v when rc = 0 do write  
end
```