

Задача №3

Павиани пресичат каньон

(70 точки)

Адаптация по

<http://www.csl.mtu.edu/cs4411/www/PROG/PG3/prog3.html>

<http://www.cs.xu.edu/csci320/06f/Assignments/JavaConcurrencyProblems.html>

Tanenbaum A. S.: *Modern Operating Systems*, 2 ed; problems 2.28, 2.29

Описание на задачата

През територията на колония от павиани преминава дълбок каньон. За да преминават от едната на другата страна на каньона павианите използват въже, което е опънато над каньона от единия до другия край. Въжето е достатъчно здраво за да издържи произволен брой павиани, които пресичат едновременно в една и съща посока от едната към другата страна на каньона. Но въжето е тънко и не позволява на павианите да се разминават при пресичане на каньона. Поради това един павиан, който иска да пресече каньона от изток на запад трябва да изчака павианите, които се движат по въжето от запад на изток да пресекат и да освободят въжето.

Ако въжето се използва от павиан, който се движи от изток на запад, то другите павиани от източната страна на каньона могат да започнат да пресичат каньона независимо от това колко павиана чакат от западната страна на каньона. Този протокол за преминаване по въжето може да доведе до *изгладняване (starvation)*.

За да се справят с този проблем приели следния *протокол* за преминаване през каньона: когато даден павиан **A** иска да пресече каньона на изток (съответно на запад), той застава до въжето. Ако в момента въжето е заето от друг павиан **B**, който се движи на запад (съответно на изток), то павиан **A** изчаква докато въжето се освободи; в това време павианите движещи се на запад не могат да използват въжето, докато поне един павиан (павиан **B**) движещ се на изток не пресече каньона.

Схема на решението

Поведението на всеки един от павианите трябва да се симулира с отделна нишка. Първоначално трябва да се създадат $e > 0$ павиана, които се движат на изток, и $w > 0$ павиана, които се движат на запад. Всеки павиан пресича каньона $t > 0$ пъти, след което изчезва завинаги. Всеки павиан, който се движи на изток (съответно на запад) след като пресече и достигни източната страна на каньона (съответно западната) магично се прехвърля обратно на западната страна на каньона (съответно източната) и отново започва своето пътешествие на изток (съответно на запад). Това се повтаря $t > 0$ пъти. Следващия фрагмент описва поведението на нишката, която симулира павиан, където **DIRECTION** определя посоката, в която се движи павиана:

```
for (int i = 1; i <= t; i++) {
    delay(); // играе си известно време
    getOnRope(DIRECTION, ...); // прави заявка да се качи
                                // на въжето
    delay(); // павиана се е качил на въжето
```

```

// и пресича каньона; движи се по
// по въжето известно време
getOffRope(DIRECTION, ...); // слиза от въжето на другия край
// павиана е от другата страна на каньона, отпечатва съобщение
// и магично се връща обратно на изходна позиция
}

```

Изисквания към решението

Вашата програма трябва да симулира поведението на павианите както е описано по-горе. В частно програмата трябва да осигурява следното:

1. Всички павиани, които са се хванали за въжето, се движат в една и съща посока.
2. Павианите трябва да използват въжето за да пресекат каньона. С други думи павианите не могат *магично* да пресекат каньона и *магично* да се върнат обратно. Магията е приложима само при връщане.
3. Павианите не могат да се качват на въжето ако нямат право да го направят.
4. Когато един павиан иска да пресече, но по въжето се движат павиани в обратна посока, то нито един друг павиан повече не може да започне пресичане в обратна посока. По-точно, ако един павиан иска да пресече каньона на изток, а по въжето има павиани които се движат на запад, то другите павиани, които искат да пресекат каньона на запад трябва да изчакат павиана, който иска да се движи на изток.
5. Когато няколко павиана се движат по въжето в една и съща посока, те не могат да се изпреварват.

Вход

Входът на вашата програма се състои от следното: броят **e** на павианите, които пресичат каньона на изток, броят **w** на павианите, които се движат на запад, и броят **t** пресичания, които трябва да направи всеки павиан преди. Всички аргументи на програмата трябва да се прочетат от командния ред:

```
prog3 e w t
```

Например командния ред

```
prog3 15 8 12
```

означава, че програмата трябва да симулира 15 павиана, които пресичат каньона на изток, 8 павиана, които пресичат каньона на запад и че всеки от павианите трябва да пресече каньона по 12 пъти. Стойностите на тези параметри трябва да са по-големи от 0.

Изход

Вашата програма трябва да генерира изход, подобен на следния:

```

Eastward-moving baboon 3 started
Westward-moving baboon 1 started
Eastward-moving baboon 4 started
.....
Eastward-moving baboon 1 arrives at the west end
Westward-moving baboon 3 arrives at the east end
.....
Eastward-moving baboon 1 is on the rope

```

Eastward-moving baboon 3 is on the rope
Eastward-moving baboon 4 is on the rope
Westward-moving baboon 2 arrives at the east end
Eastward-moving baboon 5 arrives at the west end
Eastward-moving baboon 1 completes crossing the canyon (2)
Eastward-moving baboon 3 completes crossing the canyon (1)
Westward-moving baboon 1 arrives at the east end
Eastward-moving baboon 4 completes crossing the canyon (3)
Westward-moving baboon 2 is on the rope

Westward-moving baboon 2 completes all (xx) crossings and retires.
Bye-Bye!

Поради динамичното поведение на многонишковите приложения, изходът от вашата програма няма да бъде напълно идентичен.

Следният ред означава, че нишката, която симулира павиан номер 3 движещ се на изток, е стартиране:

Eastward-moving baboon 3 started

Следният ред означава, че павиан номер 1, движещ се на изток е достигнал до западния край на каньона:

Eastward-moving baboon 1 arrives at the west end

Следния ред означава, че павиан номер 4 е завършил пресичането на каньона. Числото в края на съобщението показва броя пресичания, които е изпълнил павиана.

Eastward-moving baboon 4 completes crossing the canyon (3)

Следния ред означава, че павиан номер 3, движещ се на изток, се е качил на въжето:

Eastward-moving baboon 3 is on the rope

Следния фрагмент от изхода илюстрира важна част от решението на задачата. След като павианите движещи се на изток с номера 1, 3 и 4 са се качили на въжето, павиан, движещ се на запад с номер 2 застава на източния край на въжето. Макар че павиан номер 5, движещ се на изток, застава на западния край на въжето, той не започва да пресича каньона, защото е пристигнал след павиан 2, движещ се на запад. След като павианите 1, 3 и 4, движещи се на изток, стигнат до източния край на каньона, павиан номер 2, движещ се на запад, може да се качи на въжето.

Eastward-moving baboon 1 is on the rope
Eastward-moving baboon 3 is on the rope
Eastward-moving baboon 4 is on the rope
Westward-moving baboon 2 arrives at the east end
Eastward-moving baboon 1 completes crossing the canyon (2)
Eastward-moving baboon 5 arrives at the west end
Eastward-moving baboon 3 completes crossing the canyon (1)
Westward-moving baboon 1 arrives at the east end
Eastward-moving baboon 4 completes crossing the canyon (3)
Westward-moving baboon 2 is on the rope

Следния ред показва, че павиан номер 2, движещ се на запад е изпълнил необходимия

брой пресичания на каньона и се пенсионира:

Westward-moving baboon 2 completes all (xx) crossings and retires.

Bye-Bye!