

PPaDS Zadanie5 – Jakub Trstenský 92378

1. Kombinácia synchronizačných problémov

Ide o úlohu Hodujúcich divochov, ktorá je založená na myšlienke Producentov a konzumentov, rozšírená o sledovanie stavu „skladu“ (koľko porcií sa nachádza v hrnci).

Štandardné riešenie Hodujúcich divochov však treba ešte zmeniť `empty_pot` semafor na multiplex, pretože chceme, aby začali variť všetci kuchári a naplnili hrniec doplna. Následne je potrebné vyriešiť problém, že kuchári menia a čítajú zdieľanú premennú a preto je potrebné dočasné zablokovanie prístupu k tejto premennej. Je potrebné využiť mutex.

2. Pseudokód riešenia

```
1  # premenna ozn. maximalny pocet porcii, ktore sa zmestia do hrnca
2  max_servings = N
3
4  def init():
5      mutex := Mutex()
6      servings := 0
7      fullPot := Semaphore(0)
8      emptyPot := Semaphore(0)
9
10     barrier1 := SimpleBarrier()
11     barrier2 := SimpleBarrier()
12
13     for savage_id in [0, 1, 2, ..., N-1]:
14         create_and_run_thread(savage, savage_id)
15     for cook_id in [0, 1, 2, ..., N-1]:
16         create_and_run_thread(cook, cook_id)
17
18     def getServicingFromPot(savage_id):
19         print("divoch %2d: beriem si porciu", savage_id)
20         servings := servings - 1
21
22     def savage(savage_id):
23         while True:
24             barrier1.wait("divoch %2d: prisiel som na veceru, uz nas je %2d",
25                           savage_id,
26                           print_each_thread = True)
27             barrier2.wait("divoch %2d: uz sme vsetci, zaciname vecerat",
28                           savage_id,
29                           print_last_thread = True)
30
31             mutex.lock()
32             print("divoch %2d: pocet zostavajucich porcii v hrnci je %2d" % (savage_id, servings))
33             if servings == 0:
34                 print("divoch %2d: budim kuchara" % savage_id)
35
36                 """
37                 max_servings je premenna oznacujuca maximalny pocet porcii v hrnci
38                 tymto som nastavil semafor na maximalny pocet porcii, aby kuchari varili
39                 tolko porcii kolko je potrebne na naplnenie celeho hrnca
40                 """
41                 emptyPot.signal(max_servings)
42                 fullPot.wait()
43             getServicingFromPot(savage_id)
44             mutex.unlock()
45
46             # konkurentne vykonavany kod
47             print("divoch %2d: hodujem" % savage_id)
48
```

```

49 def putServingsInPot(cook_id):
50     """
51     kazdy kuchar vari jednu porciu v danom case.
52     Po navareni prida porciu do hrnca. Ak je este
53     hrniec nedostatocne naplneny moze kuchar, ktory
54     uz varil varit dalsiu porciu.
55     """
56
57     # proces varenia
58     print("kuchar %2d: varim" % cook_id)
59     sleep(cas na varenie)
60     # pouzity mutex na zabranenie docastneho pristupu k zdielanej premennej
61     mutex.lock()
62     servings += 1
63     # Ak je hrniec plny, posledny kuchar oznami divochom, ze je navarene
64     if servings == max_servings:
65         print("kuchar %2d hrniec je plny" % cook_id)
66         fullPot.signal()
67     mutex.unlock()
68
69 def cook(cook_id):
70     while True:
71         """
72         Pustaju sa kuchari aby naplnili plny hrniec
73         emptyPot nastaveny na hodnotu max_servings
74         pouzite ako multiplex, aby mohli viaceri naraz varit
75         """
76         emptyPot.wait()
77         putServingsInPot(cook_id)

```