

ТЕХНОЛОГИЧНО УЧИЛИЩЕ “ЕЛЕКТРОННИ СИСТЕМИ”
ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ

ОПЕРАЦИОННИ СИСТЕМИ

Задача 2: Миньори

`elsys.os.2015@gmail.com`



7 септември 2016 г.

1 Условие на задачата

1.1 Основна функционалност

Целта на задачата е да се реализира синхронизация на нишки посредством `mutex`-и. Програмата представлява симулация на миньори, копаещи злато, и търговци, които го продават. Всеки миньор добива някакво злато и го добавя в склада. Търговецът продава златото от склада, независимо от миньорите, стига да има достатъчно злато.

Програмата започва с 1 миньор и с 1 търговец.

Изисквания:

- Действията на всеки миньор и на всеки търговец се извършват в отделна нишка. Тази нишка се създава веднъж за всеки миньор/търговец.
- Програмата следи колко злато има за продажба във всеки един момент.
- След всяко добиване на злато или продажба, миньорът/търговецът трябва да си почине 2 секунди, което се симулира чрез извикването на `sleep()`
- Всеки миньор добива по 10 кг злато за едно действие. Търговецът също продава по 10 кг на един път.
- Златото в мината е неограничено.
- Ако се опитаме да продадем злато, но складът е празен, трябва да се изведе съобщение "The warehouse is empty, cannot sell!"
- Търговците винаги се опитва да продават, независимо дали това е възможно или не. Това означава, че е възможно да се изведе няколко пъти "The warehouse is empty, cannot sell!" последователно.

След всяко действие трябва да се изведе съобщение на стандартния изход "Miner N gathered 10 gold" или съответно "Trader N sold 10 gold".

След 20 действия от всеки миньор/търговец нишката му приключва. След приключването на всички нишки трябва да се изведе количеството злато в склада и програмата да приключи.

Пример:

```
Miner 1 gathered 10 gold
Trader 1 sold 10 gold
Miner 1 gathered 10 gold
Trader 1 sold 10 gold
...
Trader 1 sold 10 gold
Trader 1 sold 10 gold
Gold: 0
```

1.2 Поддръжка на N миньора и M търговеца

На командия ред се подават две числа. Първото е броят на миньорите, второто е броят на търговците. По подразбиране имаме 1 миньор и 1 търговец.

```
./miners 3 4
```

```
Miner 3 gathered 10 gold
Miner 2 gathered 10 gold
Miner 1 gathered 10 gold
Trader 1 sold 10 gold
Trader 2 sold 10 gold
Trader 3 sold 10 gold
Miner 2 gathered 10 gold
Miner 3 gathered 10 gold
.....
The warehouse is empty, cannot sell!
The warehouse is empty, cannot sell!
The warehouse is empty, cannot sell!
The warehouse is empty, cannot sell!
The warehouse is empty, cannot sell!
The warehouse is empty, cannot sell!
Gold: 0
```

2 Изисквания към решението и оценяване

- Критерии за оценка:
 - Правилно създаване на нишка - 2 точки
 - Правилно изчакване на нишките - 1 точки
 - Правилно синхронизация на данните - 7 точки
 - Извикване на системната функция `sleep` в правилния момент - 2 точка
 - Поддръжка на произволен брой миньори и търговци - 7 точки
 - Обработка на грешки - 3 точки
 - Правилно освобождаване на използваните ресурси - 3 точки
- Програмата трябва да бъде написана на езика C съгласно ISO/IEC 9899:1999.
- Правилата за оценяване са следните. Приемаме, че напълно коректна и написана спрямо изискванията програма получава максималния брой точки — 100% или 25 точки. Ако в решението има пропуски, максималният брой точки ще бъде намален съгласно правилата описани по-долу.
- За работа с нишки трябва да се използва семейството от функции `pthread_create()`, `pthread_join()` и т.н. Неспазването на това изискване води до оценяване на задачата с 0 точки.
- Всеки файл от решението трябва да започва със следният коментар:

```
//-----
// NAME: Ivan Ivanov
// CLASS: XI a
// NUMBER: 13
// PROBLEM: #2
//-----
```
- Лош стил на програмиране и липсващи заглавни коментари ще ви костват 30%.
- Програми, които не се компилират получават 0 точки. Под „не се компилират“ се има предвид произволна причина, която може да причини неуспешна компилация, включително липсващи файлове, неправилни имена на файлове, синтактични грешки и други.

8. Програми, които се компилират, но не работят, не могат да получат повече от 50%.
Под „компилира се, но не работи“ се има предвид, че вие сте се опитали да решите проблема до известна степен, но не сте успели да направите пълно решение.
9. Безсмислени или мъгляви програми ще бъдат оценявани с 0 точки, независимо че се компилират.
10. Програми, които дават неправилни или непълни резултати ще получат не повече от 70%.