

ТЕХНОЛОГИЧНО УЧИЛИЩЕ “ЕЛЕКТРОННИ СИСТЕМИ”
ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ

ОПЕРАЦИОННИ СИСТЕМИ

Задача 3: Реализация на прототип на StarCraft III

elsys.os.2014@gmail.com



13 април 2015 г.

1 Условие на задачата

1.1 Основна функционалност (25 точки)

Студиото за разработка на игри Blizzard Entertainment има нужда от помощ в разработването на прототип на следваща игра от поредицата StarCraft. Подобно на предишните версии на играта, когато играта стартира, играчът разполага с 5 работници (SCV), Команден център (Command center) и 0 минерали. Целта на играча е да построи 20 войника, след което играта приключва. Войниците струват минерали, които се набавят с помощта на работници. Играчът подава команди на стандартния вход. Целта на задачата е да се реализира синхронизация на нишки посредством `mutex`-и.

1.1.1 Работници

Работниците служат за добиване на минерали. Изпълняват следната поредица от действия:

1. Копаене на минерали (не се симулира допълнително време).
Преди започването на копаене на стандартния изход трябва да се изведе съобщението - "SCV N is mining", където N е поредният номер на работника.
2. Транспортиране на минералите до свободен Команден център (симулира се време, 2s).
При стартиране на транспортирането трябва да се изведе - "SCV N is transporting minerals".
3. Разтоварване на минералите (не се симулира допълнително време).
След успешно разтоварване трябва да се изведе - "SCV N delivered minerals to Command center M", където M е поредният номер на Командния център.
4. (Връщане на стъпка 1).

При всяко копаене се добиват 8 минерала. Всеки работник трябва да се изпълнява в отделна нишка. При изчерпване на минералите на картата нишките на работниците трябва да прекратят изпълнението си.

1.1.2 Командни центрове

Командните центрове служат за обучение на войници и като място за разтоварване на минерали. Обучението на войници не пречи на разтоварването на минерали. Изисквания:

- В Командния център трябва да може да се разтоварват минерали и да се обучават войници едновременно.
- Само по един работник може да разтоварва минерали в конкретен Команден център по едно и също време.

Обучаването на войник струва 50 минерала и се симулира време за обучение 1s. След обучението на войник трябва да се изведе съобщението "You wanna piece of me, boy?".

1.1.3 Команди подавани от играча

Командите се подават от играча на стандартния вход. Възможните команди са:

- "m" за да се построи нов войник.

1.1.4 Общи

- На картата има 5000 минерала.
- Общият брой работници и войници не може да надвишава 200.
- Всички нишки трябва да работят паралелно една спрямо друга.

- Синхронизацията между отделните нишки трябва да става посредством `mutex`-и.
- Симулирането на време става чрез използването на `sleep()`.
- Не е нужно да се следи за ситуации, в които играчът не може да завърши успешно играта.

При приключване на играта трябва да изведе броя на първоначалните минерали на картата, оставащите минерали и общите събрани минерали от играча. При правилно решение на задачата, сумата на събраните минерали и оставащите минерали ВИНАГИ ще е равна на първоначалните минерали.

1.1.5 Примерна игра

```
SCV 1 is mining
SCV 1 is transporting minerals
SCV 1 delivered minerals to Command Center 1
SCV 2 is mining
SCV 2 is transporting minerals
SCV 2 delivered minerals to Command Center 1
SCV 3 is mining
SCV 3 is transporting minerals
SCV 3 delivered minerals to Command Center 1
SCV 4 is mining
SCV 4 is transporting minerals
SCV 4 delivered minerals to Command Center 1
m
Not enough minerals.
SCV 5 is mining
SCV 5 is transporting minerals
SCV 5 delivered minerals to Command Center 1
SCV 1 is mining
SCV 1 is transporting minerals
SCV 1 delivered minerals to Command Center 1
SCV 2 is mining
SCV 2 is transporting minerals
SCV 2 delivered minerals to Command Center 1
m
You wanna piece of me, boy?
...
```

1.2 Поддръжка на N на брой работници (25 точки)

Играчът има възможност да обучава работници от Командния център, чрез въвеждане на команда “s” на стандартния вход. Цената на един работник е 50 минерала и времето за обучение е 1s. След обучението на работник трябва да се извежда съобщението “SCV good to go, sir.”.

```
...
SCV 1 is mining
SCV 1 is transporting minerals
SCV 1 delivered minerals to Command Center 1
s
SCV 1 is mining
SCV 1 is transporting minerals
SCV good to go, sir.
SCV 2 is mining
SCV 2 is transporting minerals
```

```
SCV 1 delivered minerals to Command Center 1
SCV 2 delivered minerals to Command Center 1
...
```

1.3 Поддръжка на М на брой Командни центрове (25 точки)

Играчът има възможност да строи нови Командни центрове. Всеки нов команден център служи само за разтоварване на ресурси - не може да обучава работници и войници. След създаването на Команден център, работниците могат да доставят минерали в който и да е Команден център, стига той да е свободен. Между проверките не трябва да има забавяне (използвайте `pthread_mutex_trylock()`). Ако няма нито един свободен, работникът трябва да изчака 1s и да опита отново.

Създаването на Команден център става чрез въвеждането на командата "c". Приема се, че за построяването няма нужда да се ангажира работник. На стандартния изход трябва да се изведе "Command center M created."

Всеки Команден център струва 400 минерала и построяването му отнема 2s.

```
...
c
...
SCV 1 is mining
SCV 1 is transporting minerals
Command center 2 created.
SCV 2 is mining
SCV 2 is transporting minerals
SCV 1 delivered minerals to Command Center 2
SCV 2 delivered minerals to Command Center 1
...
```

1.4 Обработка на грешки (25 точки)

- При логически грешки от страна на играча трябва да се изведе съобщение за грешка и играта да се продължи. Пример за логическа грешка е строенето на войник, когато няма достатъчно събрани минерали.
- Ако при извикване на `pthread_create`, `pthread_mutex_init`, и т.н. възникне грешка, трябва да се изведе съобщение за грешка и програмата да се прекрати.

2 Изисквания към решението и оценяване

1. Програмата трябва да бъде написана на езика C съгласно ISO/IEC 9899:1999.
2. Правилата за оценяване са следните. Приемаме, че напълно коректна и написана спрямо изискванията програма получава максималния брой точки — 100% или 100 точки. Ако в решението има пропуски, максималният брой точки ще бъде намален съгласно правилата описани по-долу.
3. За работа с нишки трябва да се използва семейството от функции `pthread_create()`, `pthread_mutex_init()`, и т.н.
4. Задължително към файловете с решението трябва да е приложен и `Makefile`. Изпълнимият файл, който се създава по време на компилация на решението, трябва да се казва `starcraft3`.
5. При проверка на решението програмата ви ще бъде компилирани и тествана по следния начин:

```
make
./starcraft3
```

Предходната процедура ще бъде изпълнена няколко пъти с различни входни данни за да се провери дали вашата програма работи коректно. Също така програмата трябва да продължи да работи коректно, ако се изпуснат всички извиквания към `sleep()`.

6. Реализацията на програмата трябва да спазва точно изискванията описани по-горе. Всяко отклонение от изискванията ще доведе до получаване на 0 точки за съответната част от условието.
7. Работи, които са предадени по-късно от обявеното (или не са предадени), ще бъдат оценени с 0 точки.
8. Програмата ви трябва да съдържа достатъчно коментари. Оценката на решения без коментари или с недостатъчно и/или мъгляви коментари ще бъде намалена с 30%.
9. Всеки файл от решението трябва да започва със следният коментар:

```
//-----  
// NAME: Ivan Ivanov  
// CLASS: Xia  
// NUMBER: 13  
// PROBLEM: #3  
// FILE NAME: xxxxxx.yyy.zzz (unix file name)  
// FILE PURPOSE:  
// няколко реда, които описват накратко  
// предназначението на файла  
// ...  
//-----
```

Всяка функция във вашата програма трябва да включва кратко описание в следния формат:

```
//-----  
// FUNCTION: ххууzz (име на функцията)  
// предназначение на функцията  
// PARAMETERS:  
// списък с параметрите на функцията  
// и тяхното значение  
//-----
```

10. Лош стил на програмиране и липсващи заглавни коментари ще ви костват 30%.
11. 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, неправилен или липсващ `Makefile`, и т.н. Обърнете внимание, че в UNIX имената на файловете са case sensitive.
12. 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. Често срещан проблем, който спада към този случай, е че вашият `Makefile` генерира изпълним файл, но той е именуван с име, различно от очакваното (т.е. `head` в разглеждания случай).
13. Безсмислени или мъгляви програми ще бъдат оценявани с 0 точки, независимо че се компилират.
14. Програми, които дават неправилни или непълни резултати, или програми, в които изходът и/или форматирането се различава от изискванията ще получат не повече от 70%.

15. Всички наказателни точки се сумират. Например, ако вашата програма няма задължителните коментари в началото на файлове и функциите се отнемат 30%, ако няма достатъчно коментари се отнемат още 30%, компилира се, но не работи правилно — още 30%, то тогава резултатът ще бъде: $50 * (100 - 30 - 30 - 30)\% = 50 * 10\% = 5$ точки
16. Работете самостоятелно. Групи от работи, които имат твърде много прилики една с друга, ще бъдат оценявани с 0 точки.