

ИД32

Шевченко Александр Витальевич, БПИ226, вариант 20, 7 баллов

Условие задачи:

Задача об умных цветах. На клумбе растет 10 цветов, за ними непрерывно следят два садовника и поливают увядшие цветы, при этом оба садовника очень боятся полить один и тот же цветок, который еще не начал вянуть. Иначе он пропадет из-за перелива.

Создать многопроцессное приложение, моделирующее состояния цветов на клумбе и действия садовников.

Каждый цветок — отдельный процесс который следит за своим состоянием и информирует, что он полит, увядает (нужно полить), засох или перелит и сгнил (бесполезно поливать). Каждый садовник задается отдельным потоком.

4-5 баллов

1. Настройка общей памяти:

- В сегменте общей памяти будет храниться состояние каждого цветка. К этим состояниям относятся "здоровый", "увядающий", "увядший" и "перелитый".

2. Семафоры:

- С каждым цветком будет связан семафор, который управляет доступом к его определенному состоянию в общей памяти.

3. Процессы цветков:

- Каждый цветок будет представлять собой отдельный процесс, который случайным образом меняет свое состояние с "здорового" на "увядающее".

- Когда состояние цветка меняется на "увядающее", он пытается подать садовнику сигнал о помощи через общую память.

4. Потоки садовников:

- Каждый садовник будет работать в отдельном потоке в рамках одного процесса и будет постоянно проверять общую память на наличие "увядающих" цветов.
- Когда садовник найдет "увядающий" цветок, он попытается получить семафор для этого цветка. В случае успеха он будет поливать цветок, меняя его состояние на "здоровое".

5. Завершение работы программы:

- Программа обрабатывает SIGINT (прерывание работы с клавиатуры)

Результаты тестирования программы:

```
root@4d0b130eda60:/usr/src/app# gcc -o program program.c
root@4d0b130eda60:/usr/src/app# ./program
Flower 7 has started withering.
Flower 4 has started withering.
Gardener 1 is watering flower 4.
Gardener 1 is watering flower 7.
Flower 4 has started withering.
Flower 3 has started withering.
Gardener 2 is watering flower 4.
Flower 6 has started withering.
Flower 8 has started withering.
Gardener 2 is watering flower 6.
Flower 5 has started withering.
Flower 0 has started withering.
Gardener 2 is watering flower 8.
```

6-7 баллов

Общая память включает семафоры: каждая структура Flower теперь включает в себя безымянный семафор. Эта настройка требует, чтобы семафор был доступен напрямую и использовался всеми процессами, которые отображают этот сегмент общей памяти.

```
typedef struct {  
    int state; // 0 = healthy, 1 = withering, 2 = withered, 3 =  
    sem_t semaphore; // Unnamed semaphore for each flower  
} Flower;
```

Инициализация семафора: Функция `sem_init()` используется для инициализации каждого семафора непосредственно в общей памяти. Вторым аргументом 1 указывает, что семафор используется совместно процессами.

Результаты тестирования программы:

```
● root@4d0b130eda60:/usr/src/app# gcc -o program_two program_two.c  
⊗ root@4d0b130eda60:/usr/src/app# ./program_two  
Flower 7 has started withering.  
Gardener 1 is watering flower 7.  
Flower 2 has started withering.  
Gardener 2 is watering flower 2.  
Flower 5 has started withering.  
Flower 3 has started withering.  
Gardener 2 is watering flower 5.  
Flower 1 has started withering.  
Gardener 1 is watering flower 1.  
Gardener 1 is watering flower 3.  
Flower 9 has started withering.  
Flower 3 has started withering.  
Gardener 2 is watering flower 9.  
Flower 2 has started withering.
```