

Домашняя работа №5 по дисциплине «Архитектура вычислительных систем»

Греков Николай Сергеевич БПИ 204

Вариант 6

Условие задания:

Есть три процесса-курильщика и один процесс-посредник. Курильщик непрерывно скручивает сигареты и курит их. Чтобы скрутить сигарету, нужны табак, бумага и спички. У одного процесса-курильщика есть табак, у второго – бумага, а у третьего – спички. Посредник кладет на стол по два разных случайных компонента. Тот процесс-курильщик, у которого есть третий компонент, забирает компоненты со стола, скручивает сигарету и курит. Посредник дожидается, пока курильщик закончит, затем процесс повторяется. Создать многопоточное приложение, 4 моделирующее поведение курильщиков и посредника. При решении задачи использовать семафоры.

При решении были реализованы следующие объекты:

Название класса	Объем памяти(байты)	Описание
main.cpp	5 456	Точка входа в программу
Mediator.cpp	1 569	Реализация класса посредника
Mediator.h	1 234	Описание класса посредника
MyUtility.h	1 348	Класс необходимых утилит
Process.cpp	5 274	Реализация класса процесса
Process.h	907	Описание класса процесса
Semaphore.cpp	822	Реализация класса семафора
Semaphore.h	791	Описание класса семафора
Smoker.cpp	1 116	Реализация класса курильщика
Smoker.h	1 063	Описание класса курильщика
ИТОГО:	19 580	

Решение:

В методе main создается класс Process и вызывается метод processing, который работает до тех пор, пока все предметы будут не использованы. В методе processing работает семафор, который координирует действия между потоками. Во время работы программы создается три потока: два нужны для посредника и один для курильщика.

Входные параметры передаются через аргументы метода main. Там необходимо указать следующие данные: режим работы, количество пар объектов, имена курильщиков.

- *Режим работы.* <-w> – пользователь сам, через консоль будет вводить по паре элементов. <-r> – пары элементов будут случайным образом будут сгенерированы программой.
- *Количество пар объектов.* Целое неотрицательное число.
- *Имена курильщиков.* Строки через пробел.

Выбранная архитектура параллельной работы:

Под мою задачу очень хорошо подошла модель «Производитель и потребители», так как процесс-курильщики ожидают, когда процесс-посредник достанет объекты и те его смогу взять. То есть образуется конвейер.