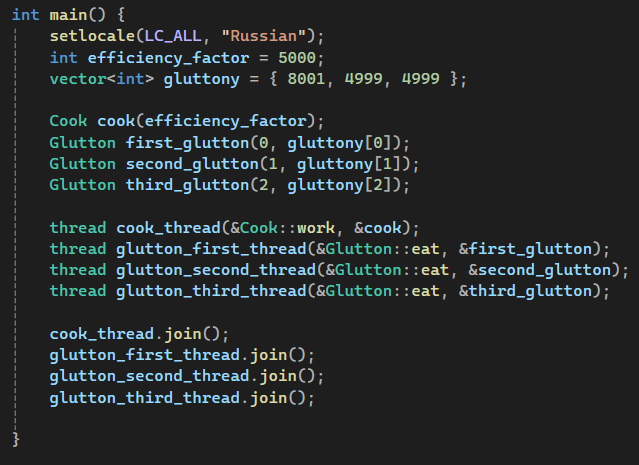
Лабораторная работа по СРВ №6

“Три толстяка”

Морозов Андрей 3.3 МО

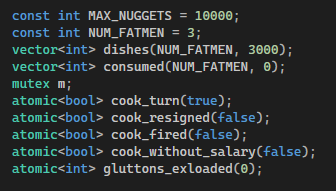
*1. Описание реализации на языке C++*

**Функция main. Инициализация параметров героев, создание потоков**



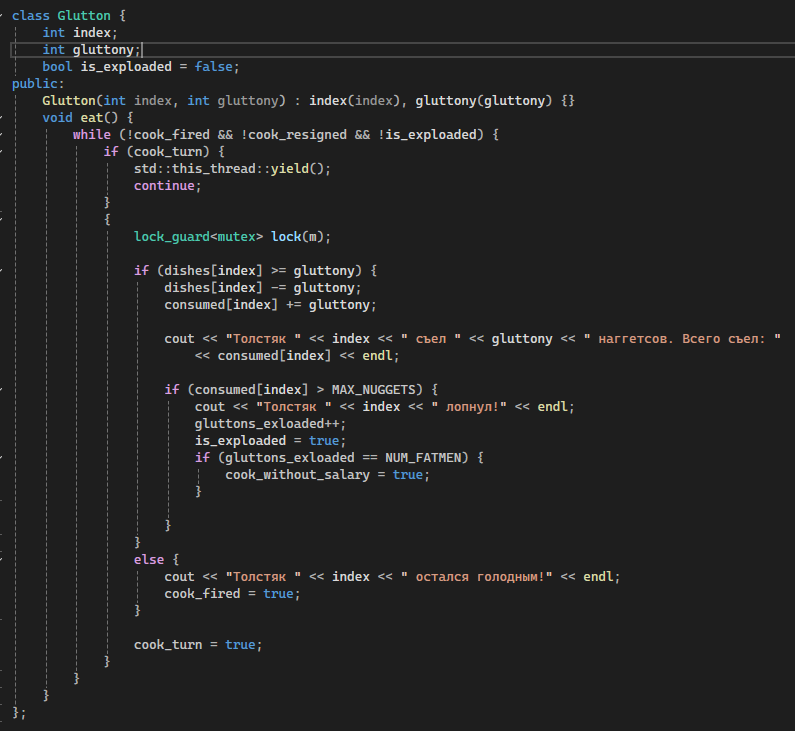
Первым делом определяем параметры героев (толстяки, кук). Производительность повара определяю в отдельной переменной, а для обжор - в векторе, где значения соотносятся с толстяками по индексам с нулевого по второй включительно.

**Переменные глобальной области видимости и их стандартные значения.**



По условию создаём три тарелки по 3000 наггетсов в каждой, граница съеденных наггетсов, число толстяков. consumed - вектор int значений - сколько съел каждый толстяк. Флаги: очередь кука раскладывать наггетсы, уволился / уволен ли кук, лишен ли зарплаты (если все толстяки взорвались). А также число толстяков, переевших наггетсы. Использую atimic с тем чтобы избежать промежуточных состояний операций переключения флагов - атомарные операции однозначно будут либо выполнены, либо нет. В контексте задачи, когда у нас у каждого толстяка по потоку + поток для кука и все они обращаются к флагам и меняют их значения, важно учитывать атомарность.

**Класс толстяка. Переменные. Функция eat**



Поля: индекс - для вывода в консоль, чтения и записи по нему значений из векторов, коэффициент обжорства, факт того, что толстяк взорвался (false при создании)

Метод eat(): в цикле while, если кук ещё работает и толстяк не взорвался, первым делом проверяем, не раскладывает ли кук наггетсы, “чтобы не откусить руку”. В случае cook\_turn == true используем yield() с тем чтобы передвинуть данный поток среди остальных потоков в очереди одного приоритета назад. Иначе, поток захватывает мьютекс, до тех пор, пока находимся в области видимости создания lock\_guard<mutex> lock(m) - с этой целью и оградил его двумя фигурными скобками.

Далее проверки на наличие наггетсов не менее коэф-та обжорства (иначе увольняем кука). Толстяк кушает, возможно, взрывается, и, если взорвался третий по счёту, кук не получает ЗП.

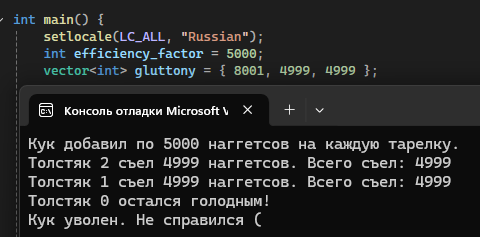
**Класс кука**



В целом, аналогичен классу толстяка с теми отличиями, что происходит подсчет дней, чтобы уволится после пятого, в цикле for в скоупе создания lock\_guard кук раскладывает наггетсы.

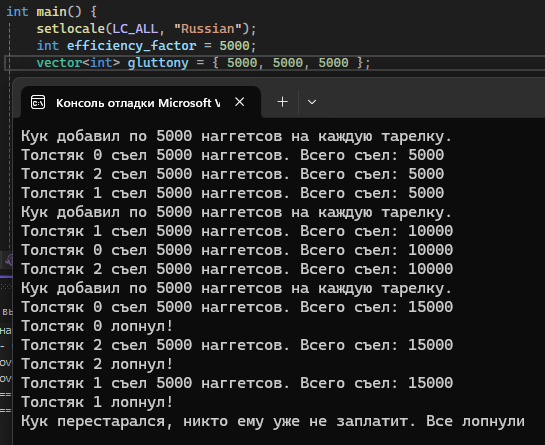
*2. Рассмотрение исходов при различных коэффициентах обжорства / производительности*

2.1 Кук уволен



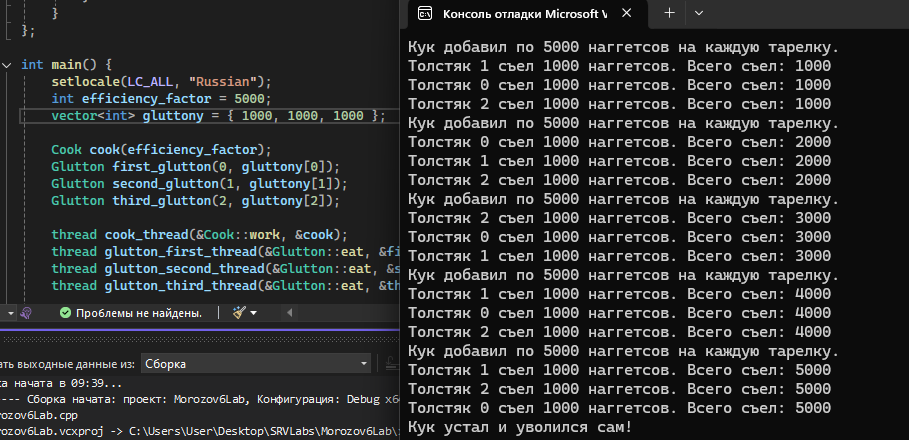
У первый толстяк съел и 3000 начальных, и 5000 выданных куком наггетсов. Не хватило одного - кук уволен.

2.2 Кук не получил зарплату



Каждый день все кушали по 5000 наггетсов. Был достигнут лимит 10000 для каждого на третий день

2.3 Кук уволился

Никто не был голоден, прошло пять дней