Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Московский государственный технический университет имени Н.Э. Баумана»

(МГТУ им. Н.Э. Баумана)

Факультет: Информатика и системы управления

Кафедра: Теоретическая информатика и компьютерные технологии

**Лабораторная работа №4**

**«Задача о пяти обедающих философах»**

**по курсу: «Разработка параллельных и распределенных программ»**

Выполнил:

Студент группы ИУ9-52Б

Марченко А.И.

Проверил:

Царев А.С.

**1. Цель работы**

Разработать с помощью потоков программу, симулирующую процесс обеда 5 и более философов. Избежать ситуации самоблокировки.

**2. Реализация**

Код программы:

#include <iostream>  
#include <thread>  
#include <mutex>  
#include <vector>  
#include <random>  
#include <condition\_variable>  
  
const int num\_of\_philosophers = 5;  
  
std::mutex print\_lock;  
  
std::atomic <bool> meal\_flag {true};  
  
class Fork{  
public:  
 std::mutex mutex;  
 std::atomic <bool> locked = false;  
 void take\_fork() {  
 while(locked){};  
 locked = true;  
 }  
 void free(){  
 locked = false;  
 }  
};  
  
class Table{  
public:  
 std::atomic <bool> ready {false};  
 std::array <Fork, num\_of\_philosophers> forks;  
};  
  
class Philosopher {  
private:  
 Fork &lfork;  
 Fork &rfork;  
 int num{};  
 Table &table;  
 std::thread phthread;  
 std::mt19937 rd { std::random\_device{}() };  
 std::string status;  
  
  
public:  
 int num\_finished\_meals = 0;  
  
 Philosopher(int \_num, Table &\_table, Fork &l, Fork &r) :  
 num(\_num),  
 table(\_table),  
 lfork(l),  
 rfork(r),  
 phthread(&Philosopher::process, this){}  
  
 ~Philosopher(){  
 std::cout << "Killing philosopher " << std::to\_string(num) << std::endl;  
 phthread.join();  
 }  
  
 void process(){  
 while(!table.ready){}  
 while(table.ready && meal\_flag) {  
 think();  
 if (!meal\_flag) return;  
 eat();  
 }  
 }  
  
 void output(const std::string& text) const {  
 std::lock\_guard<std::mutex> cout\_lock(print\_lock);  
 std::cout << "philosopher № " << num << text << std::endl;  
 }  
  
 void eat() {  
 lfork.take\_fork();  
  
 status = "tacked left fork";  
  
 thread\_local std::uniform\_int\_distribution<> forking(1, 3);  
 std::this\_thread::sleep\_for(std::chrono::milliseconds(forking(rd) \* 100));  
   
 rfork.take\_fork();  
   
 status = "is eating";  
  
 thread\_local std::uniform\_int\_distribution <> dist (1, 3);  
 std::this\_thread::sleep\_for(std::chrono::milliseconds(dist(rd) \* 100));  
   
 status = "finished eating";  
 lfork.free();  
 rfork.free();  
 num\_finished\_meals++;  
 }  
  
 void think() {  
 thread\_local std::uniform\_int\_distribution <> dist (1, 3);  
 std::this\_thread::sleep\_for(std::chrono::milliseconds(dist(rd) \* 100));  
  
 //output(" is thinking.");  
 status = "is thinking";  
 }  
  
 std::string get\_status(){  
 return std::to\_string(num) + ": " + status + " ";  
 }  
  
};  
  
  
int main() {  
 std::cout << "Start dinner for " << num\_of\_philosophers << " guests" << std::endl;  
  
 Table table;  
 std::array <Philosopher, num\_of\_philosophers> philosophers {  
 {  
 {1, table, table.forks[0], table.forks[1]},  
 {2, table, table.forks[1], table.forks[2]},  
 {3, table, table.forks[2], table.forks[3]},  
 {4, table, table.forks[3], table.forks[4]},  
 {5, table, table.forks[0], table.forks[4]}  
 }};  
  
 table.ready = true;  
 //std::this\_thread::sleep\_for(std::chrono::seconds(15));  
  
 for (int t = 0; t < 100; t++){  
 for (auto & philosopher : philosophers){  
 std::cout << philosopher.get\_status();  
 }  
 std::cout << std::endl;  
 std::this\_thread::sleep\_for(std::chrono::milliseconds(250));  
 }  
  
 meal\_flag = false;  
 std::cout << "Dinner done." << std::endl;  
 for (auto & philosopher : philosophers){  
 std::cout << philosopher.num\_finished\_meals << std::endl;  
 }  
  
 return 0;  
}

}

**3. Лог состояний философов:**

Start dinner for 5 guests

1: 2: 3: 4: 5:

1: tacked left fork 2: tacked left fork 3: tacked left fork 4: tacked left fork 5:

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: tacked left fork 2: is eating 3: finished eating 4: is eating 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: finished eating 5: is thinking

1: is thinking 2: tacked left fork 3: is eating 4: is thinking 5: tacked left fork

1: tacked left fork 2: is eating 3: is thinking 4: is eating 5: finished eating

1: tacked left fork 2: finished eating 3: tacked left fork 4: tacked left fork 5: finished eating

1: is eating 2: is thinking 3: tacked left fork 4: tacked left fork 5: is thinking

1: finished eating 2: tacked left fork 3: is eating 4: finished eating 5: tacked left fork

1: is thinking 2: is eating 3: finished eating 4: tacked left fork 5: is eating

1: tacked left fork 2: is eating 3: finished eating 4: finished eating 5: finished eating

1: is eating 2: finished eating 3: tacked left fork 4: tacked left fork 5: is thinking

1: finished eating 2: tacked left fork 3: tacked left fork 4: tacked left fork 5: tacked left fork

1: tacked left fork 2: tacked left fork 3: tacked left fork 4: is eating 5: finished eating

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: tacked left fork 5: is thinking

1: finished eating 2: finished eating 3: is eating 4: finished eating 5: tacked left fork

1: tacked left fork 2: finished eating 3: tacked left fork 4: is eating 5: finished eating

1: is eating 2: is thinking 3: is eating 4: finished eating 5: is thinking

1: is thinking 2: tacked left fork 3: tacked left fork 4: tacked left fork 5: is eating

1: tacked left fork 2: tacked left fork 3: is eating 4: is thinking 5: is thinking

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: finished eating 5: is thinking

1: finished eating 2: tacked left fork 3: finished eating 4: tacked left fork 5: is eating

1: tacked left fork 2: finished eating 3: tacked left fork 4: is eating 5: finished eating

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: tacked left fork 2: is eating 3: finished eating 4: finished eating 5: is thinking

1: finished eating 2: tacked left fork 3: tacked left fork 4: is eating 5: tacked left fork

1: finished eating 2: tacked left fork 3: finished eating 4: tacked left fork 5: is eating

1: is eating 2: finished eating 3: tacked left fork 4: is eating 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: tacked left fork 5: is thinking

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: is eating

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: finished eating

1: is eating 2: is thinking 3: tacked left fork 4: is eating 5: is thinking

1: finished eating 2: tacked left fork 3: tacked left fork 4: is eating 5: tacked left fork

1: tacked left fork 2: tacked left fork 3: is eating 4: is thinking 5: finished eating

1: is eating 2: finished eating 3: finished eating 4: tacked left fork 5: is thinking

1: finished eating 2: tacked left fork 3: tacked left fork 4: is eating 5: tacked left fork

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: is eating

1: tacked left fork 2: is eating 3: is thinking 4: is eating 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: is eating 5: is thinking

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: is eating

1: is thinking 2: tacked left fork 3: finished eating 4: tacked left fork 5: is eating

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: is eating 5: is thinking

1: finished eating 2: tacked left fork 3: tacked left fork 4: finished eating 5: is eating

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: finished eating

1: tacked left fork 2: is eating 3: finished eating 4: is eating 5: is thinking

1: finished eating 2: tacked left fork 3: is eating 4: is thinking 5: tacked left fork

1: finished eating 2: tacked left fork 3: finished eating 4: tacked left fork 5: tacked left fork

1: tacked left fork 2: is eating 3: is thinking 4: is eating 5: finished eating

1: tacked left fork 2: finished eating 3: tacked left fork 4: finished eating 5: is thinking

1: finished eating 2: tacked left fork 3: finished eating 4: tacked left fork 5: tacked left fork

1: is thinking 2: is eating 3: is thinking 4: is eating 5: tacked left fork

1: is thinking 2: finished eating 3: tacked left fork 4: tacked left fork 5: is eating

1: tacked left fork 2: tacked left fork 3: tacked left fork 4: is eating 5: finished eating

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: tacked left fork 2: tacked left fork 3: is eating 4: is thinking 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: tacked left fork 5: is thinking

1: finished eating 2: tacked left fork 3: tacked left fork 4: is eating 5: tacked left fork

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: is eating

1: tacked left fork 2: finished eating 3: finished eating 4: tacked left fork 5: is thinking

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: tacked left fork

1: is thinking 2: is eating 3: finished eating 4: tacked left fork 5: is eating

1: tacked left fork 2: finished eating 3: tacked left fork 4: is eating 5: finished eating

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: is eating 2: finished eating 3: finished eating 4: tacked left fork 5: is thinking

1: is eating 2: is thinking 3: tacked left fork 4: is eating 5: is thinking

1: tacked left fork 2: tacked left fork 3: finished eating 4: tacked left fork 5: finished eating

1: is eating 2: finished eating 3: finished eating 4: is eating 5: is thinking

1: is thinking 2: tacked left fork 3: tacked left fork 4: tacked left fork 5: tacked left fork

1: is thinking 2: tacked left fork 3: tacked left fork 4: tacked left fork 5: is eating

1: tacked left fork 2: tacked left fork 3: tacked left fork 4: is eating 5: is thinking

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: finished eating 2: finished eating 3: tacked left fork 4: tacked left fork 5: tacked left fork

1: is thinking 2: tacked left fork 3: tacked left fork 4: is eating 5: tacked left fork

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: finished eating

1: is eating 2: finished eating 3: tacked left fork 4: is eating 5: is thinking

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: tacked left fork

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: finished eating

1: is eating 2: is thinking 3: tacked left fork 4: is eating 5: is thinking

1: is eating 2: is thinking 3: finished eating 4: finished eating 5: is thinking

1: is thinking 2: tacked left fork 3: is eating 4: finished eating 5: tacked left fork

1: tacked left fork 2: tacked left fork 3: is eating 4: is thinking 5: finished eating

1: is eating 2: finished eating 3: tacked left fork 4: is eating 5: is thinking

1: finished eating 2: tacked left fork 3: is eating 4: finished eating 5: tacked left fork

1: is thinking 2: is eating 3: finished eating 4: tacked left fork 5: is eating

1: tacked left fork 2: finished eating 3: tacked left fork 4: is eating 5: finished eating

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: is thinking

1: is eating 2: finished eating 3: tacked left fork 4: is eating 5: is thinking

1: finished eating 2: tacked left fork 3: finished eating 4: finished eating 5: tacked left fork

1: is thinking 2: is eating 3: is thinking 4: is eating 5: tacked left fork

1: tacked left fork 2: finished eating 3: tacked left fork 4: finished eating 5: finished eating

1: tacked left fork 2: tacked left fork 3: tacked left fork 4: is eating 5: is thinking

1: tacked left fork 2: tacked left fork 3: is eating 4: finished eating 5: is thinking

1: tacked left fork 2: is eating 3: finished eating 4: tacked left fork 5: is thinking

1: finished eating 2: finished eating 3: tacked left fork 4: is eating 5: tacked left fork

Dinner done.

24

29

32

39

24

Killing philosopher 5

Killing philosopher 4

Killing philosopher 3

Killing philosopher 2

Killing philosopher 1

Process finished with exit code 0

Видим, что никакие соседние философы не едят одновременно, при этом ситуаций взаимоблокировки не возникает.