#include <iostream>

#include <Windows.h>

#include <thread>

#include <mutex>

#include <chrono>

using namespace std;

mutex mtx1, mtx2, mtx3;

void first()

{

mtx1.lock();

cout << "1 -ый курит " << endl;

this\_thread::sleep\_for(chrono::milliseconds(100));

mtx1.unlock();

}

void second()

{

mtx2.lock();

cout << "2 -ой курит" << endl;

this\_thread::sleep\_for(chrono::milliseconds(100));

mtx2.unlock();

}

void three()

{

mtx3.lock();

cout << "3 -ий курит" << endl;

this\_thread::sleep\_for(chrono::milliseconds(100));

mtx3.unlock();

}

int main()

{

setlocale(LC\_ALL, "ru");

mtx1.lock();

mtx2.lock();

mtx3.lock();

std::thread t1(first);

std::thread t2(second);

std::thread t3(three);

while (1)

{

int d = 1 + rand() % 3;

//cout << d << endl;

switch (d)

{

case 1:

t1.detach();

cout << "На столе: бумага, спички" << endl;

mtx1.unlock();

this\_thread::sleep\_for(chrono::milliseconds(500));

mtx1.lock();

t1 = thread(first);

break;

case 2:

t2.detach();

cout << "На столе: табак, спички" << endl;

mtx2.unlock();

this\_thread::sleep\_for(chrono::milliseconds(500));

mtx2.lock();

t2 = thread(second);

break;

case 3:

t3.detach();

cout << "На столе: табак, бумага" << endl;

mtx3.unlock();

this\_thread::sleep\_for(chrono::milliseconds(500));

mtx3.lock();

t3 = thread(three);

break;

default:

break;

}

Sleep(1000);

}

return 0;

}