



Государственное образовательное учреждение высшего
профессионального образования
«Московский Государственный Технический Университет
имени Н.Э. Баумана»

ОТЧЕТ
По лабораторной №5
По курсу «Анализ алгоритмов»
на тему «Симуляция конвейера на
потоках»

Исполнитель

Студент:

Богунов Б.М.

Группа:

ИУ7-54

Принял

Преподаватель:

Волкова Л. Л.

Исв. № подл.	Подп. и дата	Подп. и дата	Подп. и дата
Исв. №	Взам. инв.	Подп. и дата	Подп. и дата

Оглавление

Постановка задачи	3
Листинг кода	4
Заключение	6

Постановка задачи

Произвести симуляцию конвейера на потоках с использованием программных средств. Произвести тесты временных характеристик алгоритмов, сделать выводы о проделанной работе.

В качестве модели конвейера выступает корабль производящий стыковку с космической станцией.

Судно может прибывать в одном из пяти состояний:

- 1) Посадка на базу
- 2) Выбор свободного ангара
- 3) Техосмотр
- 4) Заправка
- 5) Вылет с базы

Листинг кода

```
using System;
using System.Collections.Concurrent;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading;
using System.Threading.Tasks;

namespace Lab5
{
    class Program
    {
        private static ConcurrentQueue<int> LandingQueue = new ConcurrentQueue<int>(); //
Посадка
        private static ConcurrentQueue<int> HangarSelectionQueue = new ConcurrentQueue<int>();
// Выбор ангара
        private static ConcurrentQueue<int> DiagnosticsQueue = new ConcurrentQueue<int>(); //
Техосмотр
        private static ConcurrentQueue<int> FuellingQueue = new ConcurrentQueue<int>(); //
Заправка
        private static ConcurrentQueue<int> LeavingQueue = new ConcurrentQueue<int>(); //
Вылет

        private const int c = 50;

        public static void Hangar()
        {
            int ShipID;
            while (true)
            {
                if (HangarSelectionQueue.TryDequeue(out ShipID))
                {
                    Console.WriteLine("Корабль " + ShipID + " ищет свободный ангар.");
                    DiagnosticsQueue.Enqueue(ShipID);
                }
            }
        }

        public static void Diagnostics()
        {
            int ShipID;
            while (true)
            {
                if (DiagnosticsQueue.TryDequeue(out ShipID))
                {
                    Console.WriteLine("Корабль " + ShipID + " производит техосмотр.");
                    FuellingQueue.Enqueue(ShipID);
                }
            }
        }

        public static void Fuelling()
        {
            int ShipID;
            while (true)
            {
                if (FuellingQueue.TryDequeue(out ShipID))
                {
                    Console.WriteLine("Корабль " + ShipID + " заправляется.");
                    LeavingQueue.Enqueue(ShipID);
                }
            }
        }
    }
}
```

```

    }
}

public static void Leaving()
{
    int ShipID, count = 0;
    while (count < c)
    {
        if (LeavingQueue.TryDequeue(out ShipID))
        {
            Console.WriteLine("Корабль " + ShipID + " покидает станцию.");
            count++;
        }
    }
}

static void Main(string[] args)
{
    Random r = new Random();
    Thread ps = new Thread(new ThreadStart(Hangar));
    Thread o = new Thread(new ThreadStart(Diagnostics));
    Thread p = new Thread(new ThreadStart(Fuelling));
    Thread pu = new Thread(new ThreadStart(Leaving));

    for (int ships = 1; ships <= c; ships++)
    {
        LandingQueue.Enqueue(ships);
    }

    ps.Start();
    o.Start();
    p.Start();
    pu.Start();

    for (int ships = 1; ships <= c; ships++)
    {
        int randSleepTime = r.Next(1, 10);
        Thread.Sleep(randSleepTime);

        int ShipID;
        if (LandingQueue.TryDequeue(out ShipID))
        {
            Console.WriteLine("Корабль " + ShipID + " прилетает на станцию.");
            HangarSelectionQueue.Enqueue(ShipID);
        }
    }

    pu.Join();

    pu.Abort();
    p.Abort();
    o.Abort();
    ps.Abort();

    Console.ReadKey();
}
}
}

```

Заключение

В проведённой работе выполнена задача симуляции конвейера. Достигнута реализация конвейера на потоках на основании модели Корабль-Космическая станция. Задание было выполнено при помощи языка C# в среде Visual Studio 2015.