

Литература

1. Душин С.Е., Красов А.В., Кузьмин Н.Н., Пошехонов Л.Б. Численное моделирование систем управления / Учеб. пособие для студентов вузов, обучающихся по направлениям 550200, 651900 «Автоматизация и упр.» подгот. бакалавров, магистров и дипломир. Специалистов. М-во образования РФ. С.-Петерб. гос. электротехн. ун-т «ЛЭТИ»; [С. Е. Душин и др.]. – СПб.: 2003.
2. Сахаров Д.В., Мельников С.Е., Штеренберг С.И. Инфраструктура связи на крайнем севере как база для формирования единой инфосреды // Электросвязь, 2016. – № 5. – С. 18-20.
3. Никитин В.Н., Лагутенко О.И., Ковцур М.М. Обеспечение информационной безопасности ИТС // Электросвязь, 2014. – № 1. – С. 29-31
4. Волостных В.А., Штеренберг С.И., Гвоздев Ю.В. Проблемы обеспечения безопасности персональных данных в высших учебных заведениях // Информационные технологии и телекоммуникации, 2014. – Т. 2. – № 4. – С. 134-141.
5. Кузнецов И.А., Липатников В.А., Сахаров Д.В. Управление АСМК организации интегрированной структуры с прогнозированием состояния информационной безопасности // Электросвязь, 2016. – № 3. – С. 28-36.
6. Выбираем систему электронного документооборота: от видов СЭД до конкретного ПО. Интернет ресурс. <https://www.kp.ru/guide/sistemy-dokumentooborota.html> (дата обращения 22.09.2019).
7. URL <http://www.stek-trust.ru/zashchishchennyj-dokumentooborot> (дата обращения - 22.09.2019).
8. URL <https://ecm-journal.ru/post/Zashchishchennyj-dokumentooborot-Metodologicheskie-aspekty-realizacii.aspx> (дата обращения - 22.09.2019).

ИССЛЕДОВАНИЕ СКОРОСТЕЙ ВЫПОЛНЕНИЯ БАЗОВЫХ МАТЕМАТИЧЕСКИХ ЗАДАЧ ПОПУЛЯРНЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

И.В. Коровин, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича;

И.А. Пулькин, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича;

А.С. Вераян, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича.

УДК 004.431.4, 004.432

Аннотация. В статье приведено сравнение времени выполнения алгоритма первыми 10-ю языками программирования в рейтинге *TIOBE* (индекс, оценивающий популярность языков программирования на основе подсчета результатов поисковых запросов, содержащих название языка). Авторами описаны алгоритмы заполнения двумерного массива псевдослучайными четырехзначными числами, приведены результаты замеров скорости выполнений этих алгоритмов.

Ключевые слова: языки программирования; скорость выполнения ПО; Ассемблер; ООП; исполняемые файлы.

RESEARCH OF SPEEDS OF PERFORMANCE OF BASIC MATHEMATICAL PROBLEMS OF POPULAR PROGRAMMING LANGUAGES

I.V. Korovin, St. Petersburg state university of telecommunications n/a prof. M. A. Bonch-Bruevich;

I.A. Pulkin, St. Petersburg state university of telecommunications n/a prof. M. A. Bonch-Bruevich;
A.S. Veranyan, St. Petersburg state university of telecommunications n/a prof. M. A. Bonch-Bruevich.

Annotation. The article gives the comparison of the execution time of the algorithm by the first 10 programming languages in the TIOBE ranking (an index that measures the popularity of programming languages based on counting the results of search queries containing the name of the language). The authors describe the algorithms for filling a two-dimensional array with pseudo-random four-digit numbers, the results of measuring the speed of execution of these algorithms are presented.

Keywords: programming languages; software execution speed; Assembler; OOP; executable files.

Введение

В связи с развитием технологий [1, 2] появилось много языков программирования, предназначенных для самых различных задач: начиная от написания простых скриптов и заканчивая разработкой программного обеспечения (далее – ПО) для автоматизированных систем управления (далее – АСУ) и других более обширных и трудных задач. Все они имеют отличающийся синтаксис, принцип действия и скорость выполнения кода. Соответственно, встает вопрос: какой язык самый «быстрый»? Для того, чтобы ответить на этот вопрос, обратимся сперва к рейтингу языков программирования на индексе *TIOBE* (табл. 1).

Для формирования индекса используется поиск в нескольких наиболее посещаемых (по данным *Alexa*) порталах: *Google*, *Blogger*, *Wikipedia*, и других. Расчет индекса происходит ежемесячно. Методика расчета индекса очень чувствительна к внутренним алгоритмам и политикам поисковых систем, в частности, к алгоритму оценки количества найденных страниц. Например, в апреле 2004 г. после проведения в поисковой системе *Google* очистки от нечестных способов продвижения сайтов, произошло резкое падение значения индекса для языков *Java* и *C++*, хотя они и остались на вершине таблицы. Для борьбы с такими флуктуациями *TIOBE* начал использовать большее количество поисковых систем.

Таблица 1

Sep 2019	Sep 2018	Programming Language	Ratings	Change
1	1	Java	16,661%	-0,78%
2	2	C	15,205%	-0,24%
3	3	Python	9,874%	+2,22%
4	4	C++	5,635%	-1,76%
5	6	C#	3,399%	+0,10%
6	5	Visual Basic .NET	3,291%	-2,02%
7	8	JavaScript	2,128%	-0,00%
8	9	SQL	1,944%	-0,12%
9	7	PHP	1,863%	-0,91%
10	10	Objective-C	1,840%	+0,33%

Язык Ассемблера (Assembly Language)

Данный язык знаменит своей сложностью, что не удивительно – вместо привычных ключевых слов и функций из высокоуровневых языков программирования здесь прямые команды процессору. Однако наградой за большой объем кода является скорость выполнения.

Поэтому именно алгоритм, написанный на ассемблере, будет ориентиром для всего последующего исследования.

В качестве генератора псевдослучайных чисел будем использовать так называемый «Минимальный генератор Парка-Миллера» (рекуррентная последовательность, которую можно предложить для реализации генератора равномерного распределения), доработанный для получения чисел в необходимом нам диапазоне (см. листинг 1).

```
proc   WRandom
    push    edx ecx
    mov     eax,[random_seed]
    xor     edx,edx
    mov     ecx,127773
    div     ecx
    mov     ecx,eax
    mov     eax,16807
    mul     edx
    mov     edx,ecx
    mov     ecx,eax
    mov     eax,2836
    mul     edx
    sub     ecx,eax
    xor     edx,edx
    mov     eax,ecx
    mov     [random_seed],ecx
    mov     ecx,100000
    div     ecx
    mov     eax,edx
    pop     ecx edx
    ret
endp

proc   WIRandom rmin:dword,rmax:dword
    push    edx ecx
    mov     ecx,[rmax]
    sub     ecx,[rmin]
    inc     ecx
    stdcall WRandom
    xor     edx,edx
    div     ecx
    mov     eax,edx
    add     eax,[rmin]
    pop     ecx edx
    ret
endp
```

Листинг 1 – Ассемблер.

Созданный с помощью данного генератора алгоритм был запущен, как и остальные, в виртуальной машине под управлением 32-разрядной ОС *Windows 7*. Время исполнения составило 0,00763 секунды.

C#

Данный язык был создан компанией *Microsoft* специально для своего продукта *ASP.NET*, однако сейчас C# – полноценный многофункциональный язык, на котором реализуются самые разные задачи. Так как почти на всех языках в нашем списке алгоритмы заполнения массива будут примерно одинаковыми, мы приведем полный код программы только для C#, а далее – только время исполнения и комментарии (см. листинг 2).

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

namespace ScienceRandomNumsAndMassiveTHSND
{
    class Program
    {
        static void Main(string[] args)
        {
            var sw = new Stopwatch();
            Random rnd = new Random();
            int n = 1000;
            int m = 1000;
            int[,] matrix = new int[n,m];

            sw.Start();

            for (int i = 0; i < n; i++)
                for (int j = 0; j < m; j++)
                    matrix[i, j] = rnd.Next(1000, 9999);

            sw.Stop();
            Console.WriteLine(sw.Elapsed);
        }
    }
}
```

Листинг 2 – C#.

Как и *Java*, C# компилируется в промежуточный язык, который позже будет исполняться виртуальной машиной, поэтому скорость исполнения будет ниже не только языка Ассемблера, но и обычных компилируемых языков. Время исполнения составило **0,01864 с.**

C

Прародитель половины языков из списка показал быстрееший результат после ассемблера – **0,00958 с.**

C++

Результат данного языка ожидаемо очень близок к обычному Си – **0,00960 с.**

Objective-C

Как и предыдущие два, данный язык обладает хорошей скоростью исполнения – **0,00959 с.**

Python

Простой, и от того очень популярный, язык оказался достаточно медленным – **0,02078 с.** Причина кроется в том, что *Python* – интерпретируемый язык и использует промежуточный код, но в отличие от *Java* и *C#* не имеет *JIT*-компиляции.

Java

Самый популярный среди большого бизнеса язык, имеет дурную славу неповоротливого языка, однако результат оказался не сильно хуже *C#* – **0,01898 с.**

Visual Basic .NET

Данный язык после компиляции превращается в такой же промежуточный язык, что и *C#*, поэтому скорость выполнения ожидаемо почти одинаковая – **0,01873 с.**

JavaScript

Самый часто встречающийся язык программирования в повседневной жизни не смог порадовать высокой производительностью – алгоритм выполнен за **0,02135 с.**

PHP

Некогда очень популярный язык сейчас доживает свои последние годы. Одной из причин его упадка можно назвать скорость – наш алгоритм выполнен за **0,02318 с.**

Результаты исследования и сравнения представлены на рис. 1.

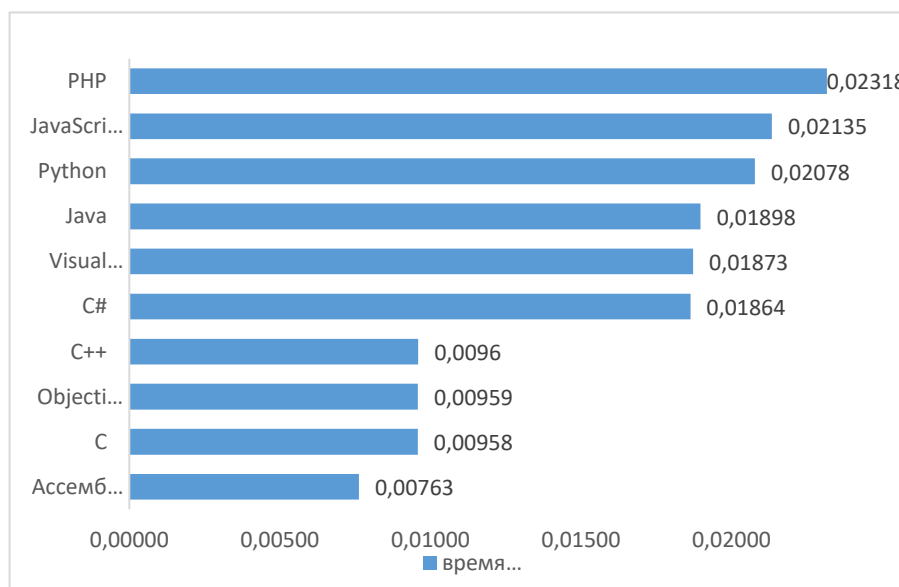


Рисунок 1

Заключение

Авторами работы представлены результаты тестирования и сравнения скоростей выполнения алгоритма различными языками программирования. Данное исследование может помочь начинающим выбрать свой первый язык программирования, а более продвинутым разработчикам – улучшить свои программы путем внедрения более эффективного кода, либо поможет с выбором следующего изучаемого языка.

Дальнейшими направлениями работы можно назвать более обширное тестирования, включая более сложные алгоритмы и большее количество исследуемых языков программирования.

Литература

1. Штеренберг С.И., Полтавцева М.А. Распределенная система обнаружения вторжений с защитой от внутреннего нарушителя // Проблемы информационной безопасности. Компьютерные системы, 2018. – № 2. – С. 59-68.
2. J. Albahari, B. Albahari – C# 7.0 in a nutshell: 2018;
3. ТЮБЕ [Электронный ресурс]. – URL: <https://www.tiobe.com/tiobe-index/>
4. Wikipedia [Электронный ресурс]. – URL: <https://ru.wikipedia.org/>
5. Cyberforum [Электронный ресурс]. – URL: <https://cyberforum.ru/>
6. StackOverflow [Электронный ресурс]. – URL: <https://stackoverflow.com/>
7. Habr [Электронный ресурс]. – URL: <https://habr.com/ru/>
8. М. Лутц – Программирование на Python. 4-е издание: 2011;
9. Г. Шилдт – Java. Полное руководство. 10-е издание: 2018;
10. Б. Страуструп – Язык программирования C++. Специальное издание: 2011;
11. I. Zhirkov – Low-Level Programming. C, Assembly, and Program Execution on Intel 64 Architecture: 2017;
12. A. W. West, S. Prettyman – Practical PHP 7, MySQL 8, and MariaDB Website Databases, 2nd edition: 2018;
13. Qaru Site [Электронный ресурс]. – URL: <http://qaru.site/>
14. Мир Кода [Электронный ресурс]. – URL: <http://www.mir-koda.ru/>
15. Практическое руководство Microsoft [Электронный ресурс]. – URL: <https://docs.microsoft.com/>

СПОСОБЫ ПРОТИВОДЕЙСТВИЯ И СРЕДСТВА РАННЕГО ВЫЯВЛЕНИЯ DDoS – АТАК

А.Н. Басканов, Московский технический университет связи и информатики, hogjam@mail.ru.

УДК 004.7

Аннотация. Изучены разновидности DDoS – атак, их влияние на работоспособность информационной системы. Представлен анализ существующих методов противодействия и раннего обнаружения DDoS – атак.

Ключевые слова: сети; DDoS; угрозы; атаки; методы противодействия; раннее обнаружение; атаки на сеть.

METHODS OF COUNTERACTION AND MEANS OF EARLY DETECTION OF DDoS - ATTACKS

Alexey Baskanov, Moscow technical university of communications and informatics.

Annotation. The varieties of DDoS attacks, their impact on the performance of the information system are studied. The analysis of existing methods of counteraction and early detection of DDoS attacks is presented.

Keywords: networks; DDoS; threats; attacks; methods of counteraction; early detection; attacks on the network.

Введение

Суть сетевой атаки заключается в нанесении злоумышленниками или программным обеспечением вреда попавшему под удар устройству. В роли наносящих вред программ