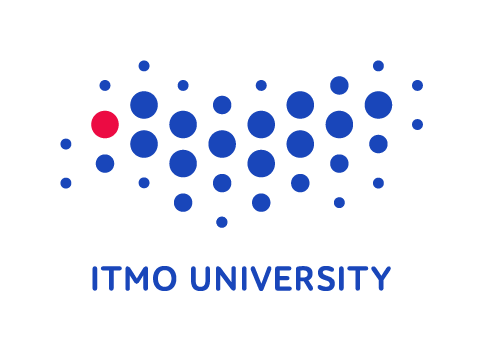
Национальный Исследовательский Университет ИТМО

Факультет Программной Инженерии и Компьютерной Техники



Вариант № 6, 5, 3 (M = 10)

Лабораторная работа № 5

по дисциплине

‘*Информатика*’

Выполнил:

Студент группы P3113

*Куперштейн Дмитрий*; : 269359

Преподаватель:

*Малышева Татьяна Алексеевна*

Санкт-Петербург 2019 г.

Оглавление

[1 Язык программирования D 3](#_Toc24213808)

[1.2 Общая характеристика и история создания 3](#_Toc24213809)

[1.3 Система типов и особенность архитектуры языка 3](#_Toc24213810)

[1.3.1 Базовые типы 3](#_Toc24213811)

[1.3.2 Производные типы 4](#_Toc24213812)

[1.3.3 Пользовательские типы 4](#_Toc24213813)

[1.3.4 Архитектура 5](#_Toc24213814)

[1.4 Примеры кода и интересные факты 5](#_Toc24213815)

[1.4.1 Hello, world! 5](#_Toc24213816)

[1.4.2 Программа, которая выводит аргументы командной строки, с которыми была вызвана 5](#_Toc24213817)

[1.4.3 Популярность языка D 5](#_Toc24213818)

[2 Язык программирования Scratch 6](#_Toc24213819)

[2.1 Общая характеристика и история создания 6](#_Toc24213820)

[2.2 Система типов и особенность архитектуры языка 6](#_Toc24213821)

[2.3 Примеры кода и интересные факты 7](#_Toc24213822)

[2.3.1 Программа, 10 раз запрашивающая строку 7](#_Toc24213823)

[2.3.2 Пример цикла с многочисленным выбором 7](#_Toc24213824)

[2.3.3 Разработчики языка Scratch 7](#_Toc24213825)

[3 Язык программирования Chapel 8](#_Toc24213826)

[3.1 Общая характеристика и история создания 8](#_Toc24213827)

[3.1.1 Система типов и особенность архитектуры языка 8](#_Toc24213828)

[3.2 Примеры кода и интересные факты 9](#_Toc24213829)

[3.2.1 Hello 9](#_Toc24213830)

[4 Список использованной литературы 10](#_Toc24213831)

## 

# Язык программирования D

## Общая характеристика и история создания

1.  ‘D — язык программирования общего назначения, предназначенный для прикладного и системного программирования. Он является языком более высокого уровня, нежели C++, но сохраняет возможность писать высокопроизводительный код и напрямую взаимодействовать с программным интерфейсом операционной системы и с оборудованием. D пригоден для написания как средних, так и крупных систем с миллионами строк исходного кода, а также хорошо подходит для ведения командной разработки. Язык D лёгок в изучении, поддерживает многие возможности в помощь программисту, а также предоставляет компилятору большое поле действий для проведения агрессивной оптимизации кода’. [1]

Рисунок 1. Логотип языка D

## Система типов и особенность архитектуры языка

‘Язык имеет богатый набор определённых типов данных и средств для определения новых типов. Типы в языке D разделяются на типы-значения и типы-ссылки.

### Базовые типы

Набор базовых типов можно разделить на следующие категории:

Таблица 1 Базовые типы языка D

| **Тип(ы)** | **Комментарий** |
| --- | --- |
| void | специальный тип для пустых значений |
| bool | логический тип |
| byte, short, int, long | знаковые целочисленные типы |
| ubyte, ushort, uint, ulong | беззнаковые целочисленные типы |
| float, double, real | типы для чисел с плавающей точкой |
| ifloat, idouble, ireal | мнимые |
| сfloat, сdouble, сreal | комплексные |
| char, wchar, dchar | знаковые (символьные) типы, обозначающие кодовые единицы кодировок UTF-8, UTF-16 и UTF-32 соответственно |

В отличие от C++ все размеры целочисленных типов определены спецификацией. То есть, тип int будет всегда размером 32 бита. Целочисленные литералы можно записывать в десятичной, двоичной (с префиксом 0b) и шестнадцатеричной (с префиксом 0x) системе счисления. Способ записи литералов в восьмеричной системе в стиле C (то есть с префиксом 0) был убран, так как такую запись легко спутать с десятичной. Если всё-таки нужно использовать восьмеричную систему, можно воспользоваться шаблоном **std.conv.octal**.

### Производные типы

Таблица 2 Производные типы языка D

|  |  |
| --- | --- |
| **Тип(ы)** | **Комментарий** |
| pointer | указатель |
| array | массив |
| associative array | ассоциативный массив |
| function | функция |
| delegate | делегат |
| string, wstring, dstring | удобные псевдонимы для неизменяемых массивов знаковых (символьных) типов immutable(char)[], immutable(wchar)[] и immutable(dchar)[], обозначающие неизменяемые (квалификатор **immutable**) строки Юникода в одной из кодировок UTF-8, UTF-16 и UTF-32 соответственно |

### Пользовательские типы

Таблица 3 Пользовательские типы языка D

|  |  |
| --- | --- |
| **Тип(ы)** | **Комментарий** |
| alias | псевдоним |
| enum | перечисление |
| struct | структура |
| union | структура |
| class | объединение |

### Архитектура

‘В предисловии к книге А. Александреску «Язык программирования D», Уолтер Брайт пишет, что начал разработку этого языка в 1999 году. Проект задумывался как реинжиниринг языка C++ с целью избавиться от наиболее существенных недостатков исходного языка и внедрить в него современные архитектурные решения. При создании языка D была сделана попытка соединить производительность компилируемых языков программирования с безопасностью и выразительностью динамических’. [2]

## Примеры кода и интересные факты

### Hello, world!

|  |
| --- |
| **import** std.stdio;  void main()  {  writeln ("Hello, world!");  } |

### Программа, которая выводит аргументы командной строки, с которыми была вызвана

|  |
| --- |
| **import** std.stdio: writefln;  void main(string[] args)  {  **foreach** (i, arg; args)  writefln("args[%d] = '%s'", i, arg);  } |

### Популярность языка D

По версии TIOBE Index язык программирования D находится по состоянию на ноябрь 2019 года на 18 месте в мире по популярности (после Visual Basic и до MATLAB)

# Язык программирования Scratch

## Общая характеристика и история создания

‘Скретч — визуальная событийно-ориентированная среда программирования, созданная для детей и подростков. Название произошло от слова scratching — техники, используемой хип-хоп-диджеями, которые крутят виниловые пластинки вперед-назад руками для того, чтобы смешивать музыкальные темы

Рисунок 2 Scratch Cat

Скретч создан как продолжение идей языка Лого и конструктора Лего. Скретч 1.4 был написан на языке Squeak, Скретч 2.0 и 3.0 ориентированы на работу онлайн. Скретч 2.0 был переписан на Flash и ActionScript. Скретч 3.0 (текущая версия) является улучшенной версией Скретч 2.0 и сделана на HTML5, используя движок WebGL, что даёт ему возможность работать на мобильных устройствах и планшетах. Скретч разрабатывается небольшой командой программистов для детей[3] в Массачусетском технологическом институте. Текущая версия — Скретч 3.0, выпущена в январе 2019 года. В 2008 году Скретч был портирован для микроконтроллерного модуля Arduino. Проект носит название S4A’. [3]

## Система типов и особенность архитектуры языка

‘В Scratch динамическая система типов, куда входят числовые, строковые и логические типы’. [4]

‘Основными компонентами скретч-программы являются объекты-спрайты. Спрайт состоит из графического представления — набора кадров-костюмов (англ. costume) и сценария-скрипта. Для редактирования костюмов спрайтов в скретч встроен графический редактор (англ. Paint Editor). Действие Скретч-программы происходит на сцене (англ. stage) размером 480×360 (условных) пикселей с центром координат в середине сцены.

Для программирования сценариев в Скретче используется drag-and-drop-подход: блоки из палитры блоков перетаскиваются в область скриптов’ [3]

## Примеры кода и интересные факты

### Программа, 10 раз запрашивающая строку



Рисунок 3 Пример программы на Scratch

### Пример цикла с многочисленным выбором

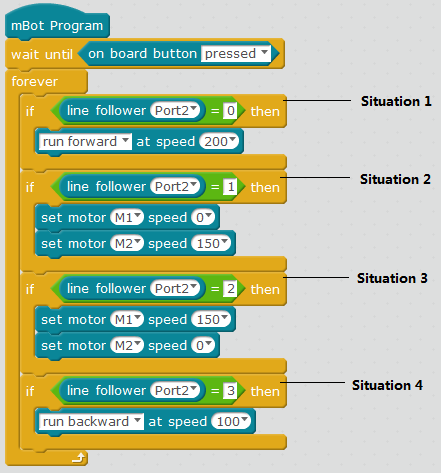


Рисунок 4 Пример программы на Scratch

### Разработчики языка Scratch

Scratch разработан и развивается командой ученых и программистов в Массачусетском технологическом институте – одном из ведущих технических вузов мира.

# Язык программирования Chapel

## Общая характеристика и история создания

‘Chapel — новый язык программирования с поддержкой распараллеливания, разработанный корпорацией Cray. Язык был разработан в рамках проекта «Каскад» (Cascade), для участия в программе DARPA Высокопродуктивные компьютерные системы (High Productivity Computing Systems, HPCS), целью которой являлось увеличение производительности суперкомпьютеров к 2010 году. Chapel был призван улучшить программируемость распараллеливания вычислений как в целом, так и для систем «Каскада» в частности, путём реализации наиболее высокого уровня выражения, нежели это позволяют нынешние языки, а также за счёт улучшения разделения на алгоритмические выражения и реализацию структур данных.’ [5]

### Система типов и особенность архитектуры языка

‘Типы данных Chapel:

* При выполнении операций передачи сообщений для указания передаваемых или получаемых данных в функциях MPI необходимо указывать тип пересылаемых данных
* MPI содержит большой набор базовых типов данных, во многом совпадающих с типами данных в алгоритмических языках C и Fortran
* В MPI имеются возможности для создания новых производных типов данных для более точного и краткого описания содержимого пересылаемых сообщений’ [6]

‘Chapel также именуемый как Каскадный высокопроизводительный язык (Cascade High Productivity Language), поддерживает модель высокоуровневого многопоточного параллельного программирования за счёт поддержки абстрагирования распараллеливания данных, задач и вложенных подзадач. Это позволяет оптимизировать локальность (компактность) данных и вычисление в программе через абстрагирование передачи данных и вложенных вычислений, определяемых данными. Язык позволяет повторно использовать и обобщать код за счёт объектно-ориентированных концепций и особенностей обобщённого программирования. Хотя Chapel и позаимствовал концепции из множества предшествующих языков, его концепции параллелизма очень близки к базовым идеям High-Performance Fortran (HPF), ZPL и расширениям языков Фортран и Си, реализованным в компьютере Cray MTA.’ [5]

## Примеры кода и интересные факты

### Hello

|  |
| --- |
| // Simple hello world  writeln("Hello, world!"); // print 'Hello, world!' to the console |

# Список использованной литературы

1. ВикиУчебник [Электронный ресурс] // Язык программирования D: https://ru.wikibooks.org/wiki/Язык\_программирования\_D
2. Википедия [Свободная энциклопедия] // D (язык программирования): https://ru.wikipedia.org/wiki/D\_(язык\_программирования)
3. Википедия [Свободная энциклопедия] // Скретч (язык программирования): https://ru.wikipedia.org/wiki/Скретч\_(язык\_программирования)
4. ИнфоУрок [Ведущий образовательный портал России] // Типы данных в Scratch: https://infourok.ru/tipi-dannih-v-scratc-2322402.html
5. Википедия [Свободная энциклопедия] // Chapel (язык программирования): https://ru.wikipedia.org/wiki/Chapel\_(язык\_программирования)
6. НГУ им. Лобачевского [Презентация] // Язык параллельного программироания Chapel: http://www.hpcc.unn.ru/file.php?id=314