Язык C# как современная альтернатива Паскалю и C++ для обучения основам алгоритмизации и программирования

Павловская Татьяна Александровна

профессор кафедры информатики и прикладной математики Санкт-Петербургского государственного университета информационных технологий, механики и оптики (СПбГУ ИТМО),

профессор кафедры информатики Санкт-Петербургского государственного университета экономики и финансов (СПбГУЭФ)

e-mail: pta-ipm@yandex.ru сайт: http://pta-ipm.narod.ru

раб. тел.: (812)233-4690, дом. тел.: (812)656-9400

План лекции

- Критерии выбора языка для обучения программированию в вузе.
- Основные характеристики и особенности C#.
- План курса и состав учебно-методического комплекса: учебник, тесты, задания, презентации лекций.
- Балльно-рейтинговая система оценки результатов обучения.
- Краткий обзор языка C# (по презентациям лекций для направления 230100 – Информатика и выч. техника, дисциплина «Программирование на языке высокого уровня»)

Учебники и учебные пособия

- Павловская Т.А. С/С++. Программирование на языке высокого уровня. — СПб.: Питер, 2001-2006. — 461 с.
- Павловская Т.А., Щупак Ю.А. С/С++. Структурное программирование: Практикум. — СПб: ПИТЕР, 2002-2006. — 240 с.
- Павловская Т.А., Щупак Ю.А. С++. Объектно-ориентированное программирование: Практикум. — СПб: ПИТЕР, 2004-2006. — 265 с.
- Павловская Т.А. Паскаль. Программирование на языке высокого уровня. Учебник — СПб.: ПИТЕР, 2003. — 384 с.
- Павловская Т.А. Паскаль. Программирование на языке высокого уровня. Практикум — СПб.: ПИТЕР, 2005. — 317 с.
- **Павловская Т.А. C#. Программирование на языке высокого уровня. — СПб.: ПИТЕР, 2006. — 432 с.**
- **http://pta-ipm.narod.ru** http://ips.ifmo.ru,
- http://de.ifmo.ru http://de.ifmo.ru, http://cis.ifmo.ru

Internet Programming school - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address http://ips.ifmo.ru/courses/pascal/index.html

Links Я Почта Я YP Новые компоненты Ср-е авто СПбГУ .NET C# MS ECMA C# СПбГУ ТПО ДОКИ C# C#-Питер

Internet Programming School

Вход | Карта сайта | Контакты

Distant learning of software development experts

ГЛАВНАЯ О ПРОЕКТЕ УЧЕБНЫЕ КУРСЫ ОЛИМПИАДЫ МОЙ IPS

- КАК УЧИТЬСЯ?
- АВТОРЫ КУРСОВ
- КУРС I
- КУРС II
  - Аннотация
  - Тема 1
  - Тема 2
  - Тема 3
  - Тема 4
  - Тема 5
  - Тема 6
  - Тема 7
  - Задачи
  - Справочные материалы
- КУРС III
- КУРС VI

ИТМО

Учебные курсы » Язык программирования Pascal

- Аннотация
- Тема 1. Основные понятия языка
  - Лекция 1. Состав языка
  - Лекция 2. Тип данных
  - Лекция 3. Линейные программы
  - Лекция 4. Процедуры ввода-вывода
- Тема 2. Управляющие операторы языка
  - Лекция 5. Операторы ветвления
  - Лекция 6. Операторы цикла
  - Лекция 7. Процедуры передачи управления
- Тема 3. Типы данных, определяемые программистом
  - Лекция 8. Простые типы даннных
  - Лекция 9. Массивы
  - Лекция 10. Строки
  - Лекция 11. Записи и множества
  - Лекция 12. Файлы
  - Лекция 13. Бестиповые и компонентные файлы
- Тема 4. Модульное программирование
  - Лекция 14. Подпрограммы
  - Лекция 15. Виды параметров подпрограмм
  - Лекция 16. Модули
  - Лекция 17. Стандартные модули Паскаля
- Тема 5. Динамическая память
  - Лекция 18. Динамическая память
  - Лекция19. Динамические структуры данных
- Тема 6. Основные понятия ООП
  - Лекция 20. Объекты
- Тема 7. Иерархии объектов
  - Лекция 21. Наследование
  - Лекция 22. Объекты в динамической памяти

Internet

Internet Programming school - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address http://ips.ifmo.ru/courses/cpp/index.html

Links Я Почта Я YP Новые компоненты Ср-е авто СПбГУ .NET C# MS ECMA C# СПбГУ ТПО ДОКИ C# C#-Питер

Internet Programming School

Вход | Карта сайта | Контакты

Distant learning of software development experts

ГЛАВНАЯ О ПРОЕКТЕ УЧЕБНЫЕ КУРСЫ ОЛИМПИАДЫ МОЙ IPS

- КАК УЧИТЬСЯ?
- АВТОРЫ КУРСОВ
- КУРС I
- КУРС II
- КУРС III
- КУРС VI
  - Аннотация
  - Тема 1
  - Тема 2
  - Тема 3
  - Тема 4
  - Тема 5
  - Тема 6

ИТМО

Учебные курсы » Программирование на C++

- Аннотация.
- Тема 1. Основные понятия алгоритмического языка
  - Лекция 1. Состав языка
  - Лекция 2. Типы данных C++
  - Лекция 3. Структура программы
  - Лекция 4. Переменные и выражения
- Тема 2. Базовые конструкции структурного программирования
  - Лекция 5. Следование и ветвление
  - Лекция 6. Операторы цикла
  - Лекция 7. Операторы передачи управления
- Тема 3. Указатели и массивы
  - Лекция 8. Указатели
  - Лекция 9. Массивы
- Тема 4. Типы данных, определяемые пользователем
  - Лекция 10. Типы данных, определяемые пользователем
- Тема 5. Функции
  - Лекция 11. Объявление и определение функций
  - Лекция 12. Параметры функции
  - Лекция 13. Функции стандартной библиотеки
- Тема 6. Модульное программирование
  - Лекция 14. Директивы препроцессора
  - Лекция 15. Области действия идентификаторов

Copyright © 2001-2006 IPS

Internet

Internet Programming school - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address http://ips.ifmo.ru/courses/pascal/problems/index.html

Links Я Почта Я YP Новые компоненты Ср-е авто СПбГУ .NET C# MS ECMA C# СПбГУ ТПО ДОКИ C# C#-Питер

Internet Programming School

Вход | Карта сайта | Контакты

Distant learning of software development experts

ГЛАВНАЯ О ПРОЕКТЕ УЧЕБНЫЕ КУРСЫ ОЛИМПИАДЫ МОЙ IPS

- КАК УЧИТЬСЯ?
- АВТОРЫ КУРСОВ
- КУРС I
- КУРС II
  - Аннотация
  - Тема 1
  - Тема 2
  - Тема 3
  - Тема 4
  - Тема 5
  - Тема 6
  - Тема 7
  - Задачи
    - Лаб.1
    - Лаб.2
    - Лаб.3
    - Лаб.4
    - Лаб.5
    - Лаб.6
    - Лаб.8
    - Лаб.9
  - Справочные материалы
- КУРС III
- КУРС VI

ИТМО

Учебные курсы » Язык программирования Pascal » Задачи

- Лаб.1. Линейные программы
- Лаб.2. Разветвляющиеся программы
- Лаб.3. Программирование циклов
- Лаб.4. Одномерные массивы
- Лаб.5. Двумерные массивы
- Лаб.6. Записи
- Лаб.8. Динамические структуры данных
- Лаб.9. Объекты

Copyright © 2001-2006 IPS

Internet

Internet Programming school - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address http://ips.ifmo.ru/courses/pascal/problems/l1/index.html

Links Я Почта Я YP Новые компоненты Ср-е авто СПбГУ .NET C# MS ECMA C# СПбГУ ТПО ДОКИ C# C#-Питер

Internet Programming School

Вход | Карта сайта | Контакты

Distant learning of software development experts

ГЛАВНАЯ О ПРОЕКТЕ УЧЕБНЫЕ КУРСЫ ОЛИМПИАДЫ МОЙ IPS

- КАК УЧИТЬСЯ?
- АВТОРЫ КУРСОВ
- КУРС I
- КУРС II
  - Аннотация
  - Тема 1
  - Тема 2
  - Тема 3
  - Тема 4
  - Тема 5
  - Тема 6
  - Тема 7
  - Задачи
    - Лаб.1
    - Лаб.2
    - Лаб.3
    - Лаб.4
    - Лаб.5
    - Лаб.6
    - Лаб.8
    - Лаб.9
  - Справочные материалы
- КУРС III
- КУРС VI

ИТМО

Учебные курсы » Язык программирования Pascal » Задачи » Линейные программы

Задача 1. Линейные программы

Линейные программы

Задача #1. Линейные программы (Отправить)

Имя входного файла	**input.txt**
Имя выходного файла	**output.txt**
Максимальное время работы на одном тесте	**2 секунды**

Заданы две формулы:

$$z_1 = 1 - {}^1/_4 \sin^2 2\alpha + \cos 2\alpha$$

$$z_2 = \cos^2\alpha + \cos^4;\alpha$$

По значениям α посчитать z_1 и z_2.

Формат входных данных

Первая строка входного файла содержит единственное число α — значение угла.

Формат выходных данных

Выходной файл должен содержать две строки, в первой должно быть значение z_1, во второй z_2. Оба числа должны быть посчитаны с точностью 5 знаков после запятой.

Пример входного файла	Пример выходного файла
1	0.37715 0.37715

Internet

(СПбГУ ИТМО)

Последовательность изучения основ программирования

1. Введение: цели обучения, критерии качества ПО, парадигмы и технологии программирования, обоснование выбора языка*
2. Состав языка (лексемы): константы, имена, ключевые слова, знаки операций, разделители
3. Концепция типа данных. Классификации типов. Встроенные типы.
4. Переменные, операции, выражения. Организация программы
5. Операторы языка
6. Составные типы данных (массивы, записи/структуры, …)
7. Подпрограммы, модули
8. Стиль программирования и технологии разработки программ
9. Динамические структуры данных: стек, очередь, список, дерево
10. Классы/объекты: описание, наследование, вложение
11. Применение библиотек объектов

©Павловская Т.А.
(СПбГУ ИТМО)

* для ОО: понятие класса

Выбор языка для обучения программированию

- BASIC
- Паскаль
- Fortran
- C
- C++
- Java
- C#
- Delphi
- Oberon
- …

Значимость критериев зависит от цели обучения и имеющихся ресурсов

Критерии выбора:

- Эффективность обучения общим основам программирования
- Современный уровень языка
- Простота обучения
- Простота оболочки
- Удобство оболочки
- Необходимые аппаратные ресурсы
- Доступность дистрибутивов
- Распространенность
- Лицензионная чистота
- Объем первоначальных знаний для простейшей программы
- Личные предпочтения
- …

©Павловская Т.А. (СПбГУ ИТМО)

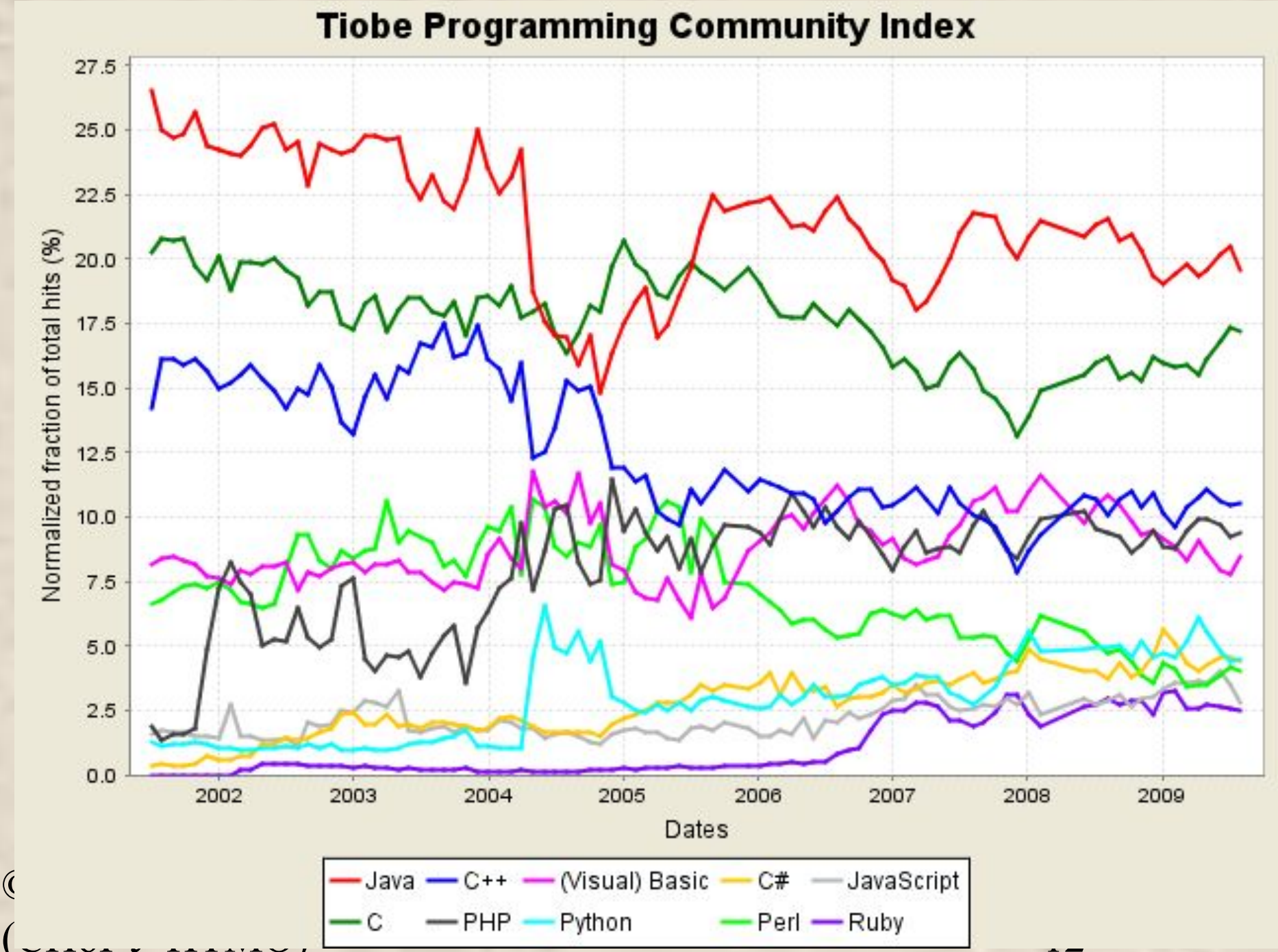
TIOBE Programming Community Index

- Отслеживает популярность языков программирования. Данные обновляются раз в месяц.
- **Рейтинг языка вычисляется исходя из количества квалифицированных кадров по всему миру, курсов и third party производителей. Для вычисления рейтинга языков программирования используются поисковые системы Google, MSN, Yahoo!**
- Рейтинг может использоваться для проверки, что ваши навыки соответствуют требованиям индустрии ПО или для принятия решения, какой язык выбрать для разработки новой системы.
- Для каждого языка программирования вычисляются рейтинг и статус. Рейтинг указывается в процентах.
- У каждого языка может быть статус "B", "A--", "A-" или "A". Языки, имеющие статус "A", считаются **основными**. Если в течение как минимум 3 месяцев рейтинг языка держится выше 0.7%, то он получает статус "A".
- Для разработки рекомендуется использовать основные языки программирования по трем причинам:
  - для неосновных языков гораздо меньше квалифицированных кадров
  - производители инструментов не занимаются разработкой инструментов для неосновных языков
  - как правило, для неосновных языков существует меньшее количество библиотек
- Конечно, это всего лишь один из критериев использования того или иного языка. Нужно учитывать также применимость языка к предметной области, производительность, масштабируемость.

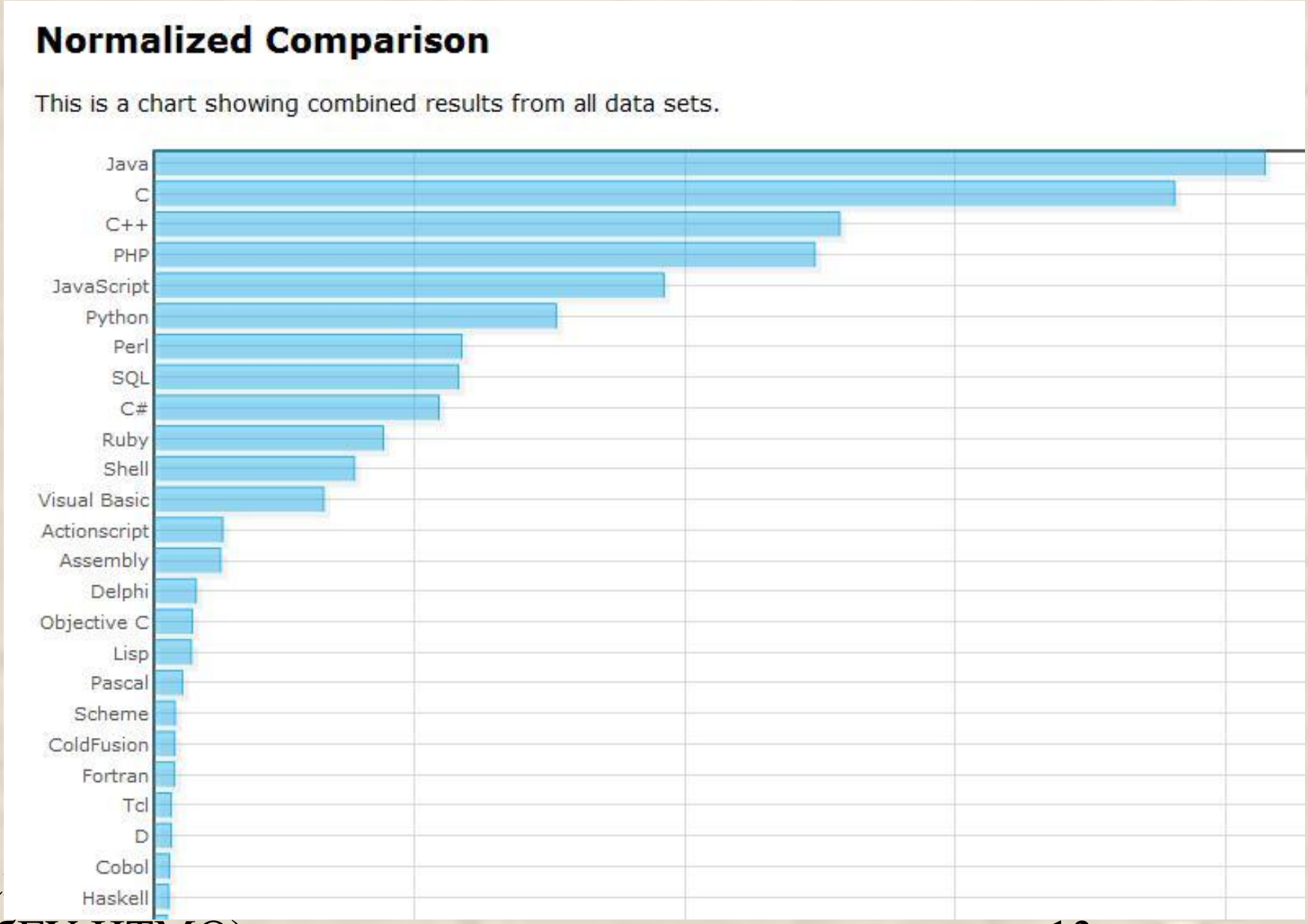
TIOBE Programming Community Index for August 2009

www.tiobe.com

Position Aug 2009	Position Aug 2008	Programming Language	Ratings Aug 2009
1	1	Java	19.527%
2	2	C	17.220%
3	4	C++	10.501%
4	5	PHP	9.390%
5	3	(Visual) Basic	8.486%
6	6	Python	4.489%
7	8	C#	4.443%
8	7	Perl	4.028%
9	10	JavaScript	2.812%
10	9	Ruby	2.490%
11	11	Delphi	2.337%


Tiobe Programming Community Index
Normalized fraction of total hits (%)
27.5
25.0
22.5
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
2002
2003
2004
2005
2006
2007
2008
2009
Dates
Java
C++
(Visual) Basic
C#
JavaScript
C
PHP
Python
Perl
Ruby

Популярность языков с langpop.com

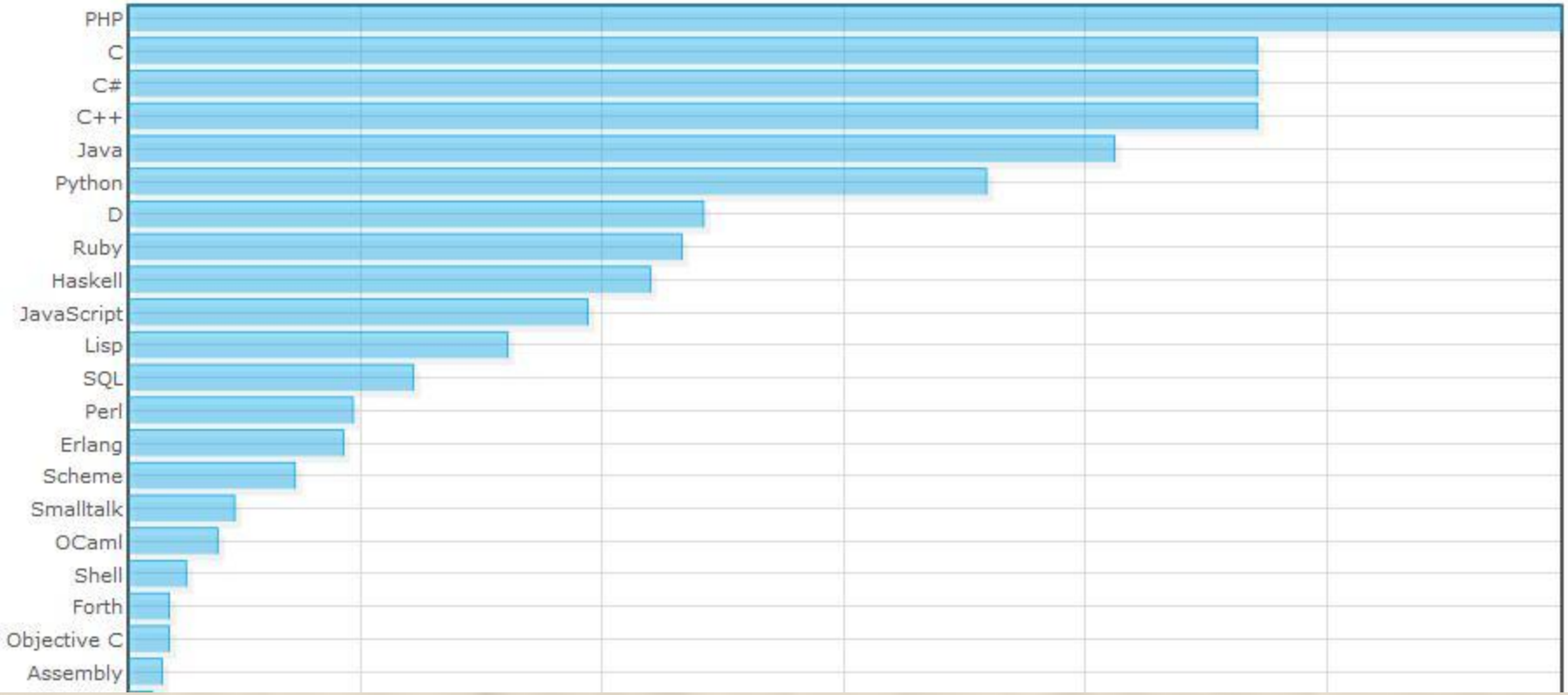


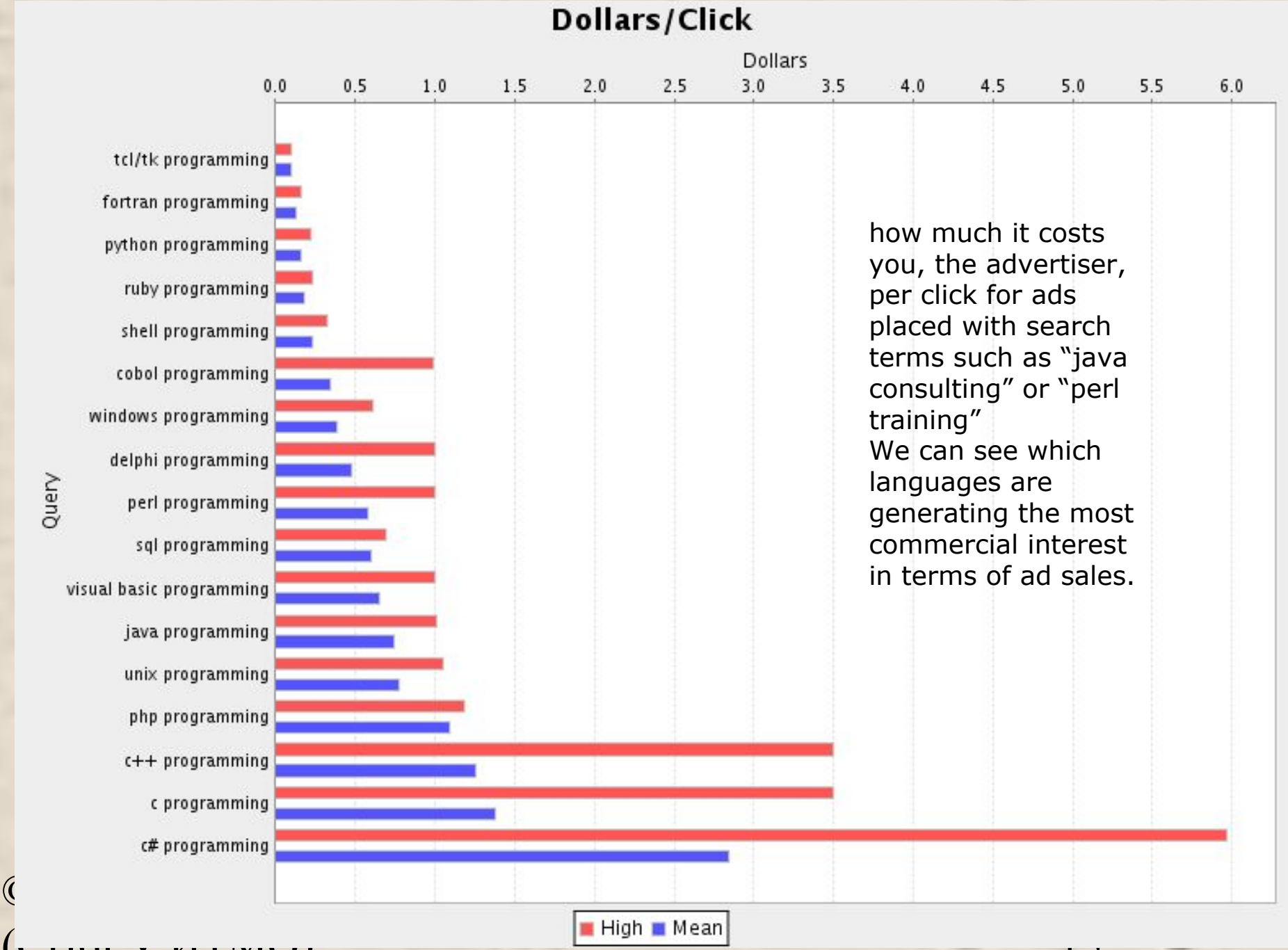
Другой срез

programming.reddit.com

The data were obtained using Yahoo's search API with the programming.reddit.com web site, and the `title:` query option, due to the `(c) 2007` at the bottom of every reddit page that returns lots of false positives for C.

This site has gained in popularity recently, and often has decent discussions of programming languages and their relative merits. The community is generally curious about up and coming languages like Haskell and Erlang. Of course there are also many people working in industry with languages like Java and PHP.





how much it costs you, the advertiser, per click for ads placed with search terms such as "java consulting" or "perl training"
We can see which languages are generating the most commercial interest in terms of ad sales.

Основные черты Оберона

- Простой и ясный синтаксис, похожий на синтаксис Паскаля (не путать с Турбо Паскалем) и Модулы-2.
- Набор структурных управляющих конструкций (IF-ELSIF-...-ELSE-END, CASE, WHILE, REPEAT, LOOP)
- Удобный и простой набор базовых типов (символы и строки, короткие и длинные целые и вещественные, логический тип, множества, процедурный тип)
  - Простые и удобные правила совместимости числовых типов (SHORTINT<=INTEGER<=LONGINT<=REAL<=LONGREAL)
- Массивы (в том числе открытые и динамические), записи (расширяемые) и указатели на массивы и записи
- Полный (но не избыточный) набор средств поддержки наследования и полиморфизма: проверка и охрана (динамического) типа, региональная охрана типа
- Встроенный сборщик мусора
- Простое модульное строение со строгим (в том числе межмодульным) контролем и динамической загрузкой модулей
- Простой и строгий экспорт и импорт
- Автоматическое формирование модулей определений (интерфейсов модулей)

Состав учебно-методического комплекса

- рабочая программа
- учебник
- учебное пособие
- методические указания по выполнению лаб. работ
- методические указания по самост. работе
- электронные средства обучения и контроля:
  - конспект лекций
  - набор тестов
  - набор заданий
  - презентации
  - справочные материалы

БаРС

в **осеннем** семестре 2009/2010 уч.гг

Формы контроля	**Модуль 1**									
	Текущий контроль по точкам								Рубежный	
	1		2		3		4			
	min	max	min	max	min	max	min	max	min	max
Лабор. работы			6	10	6	10	6	10		
Тестирование в ЦДО					5	8				
Рубежный контроль									7	10
Личностные качества									2	5
Итоговая аттестация										
Баллы точки контроля	0	0	6	10	11	18	6	10	9	15
Накопление баллов	0	0	6	10	17	28	23	38	32	53

©Павловская Т.А. (СПбГУ ИТМО)

в **осеннем** семестре 2009/2010 уч.гг

Формы контроля	Модуль 2										Итоговая аттестация
	Текущий контроль по точкам								Рубежный		
	1		2		3		4				
	min	max	min	max	min	max	min	max	min	max	
Лабор. работы			6	10	6	10	6	10			
Тестирование в ЦДО					4	6					
Рубежный контроль									4	6	
Личностные качества									2	5	
Итоговая аттестация											
Баллы точки контроля	0	0	6	10	10	16	6	10	6	11	
Накопление баллов	0	0	6	10	16	26	22	36	28	47	

©Павловская Т.А.
(СПбГУ ИТМО)

в **весеннем** семестре 2009/2010 уч. гг

58

Формы контроля	Модуль 3										Модуль 4										Экз
	Текущий контроль по точкам								Рубежный		Текущий контроль по точкам								Рубежный		
	1		2		3		4				1		2		3		4				
	min	max	min	max	min	max	min	max	min	max	min	max	min	max	min	max	min	max	min	max	
Лабор. работы			6	10	6	10	6	10					6	10			6	10			
Тестирование в ЦДО					2,5	4									5	8					
Рубежный контроль									2	3									3	5	
Личностные качества									0	5									0	5	
Итоговая аттестация																					20
Баллы точки контроля	0	0	6	10	8,5	14	6	10	2	8		0	6	10	5	8	6	10	3	10	20
Накопление баллов	0	0	6	10	14,5	24	20,5	34	22,5	42		0	6	10	11	18	17	28	20	38	58

Темы лабораторных работ

1. Линейные программы
2. Разветвляющиеся вычислительные процессы
3. Организация циклов
4. Простейшие классы
5. Одномерные массивы
6. Двумерные массивы
7. Строки
8. Стандартные контейнеры
9. Классы
10. Наследование
11. Создание Windows-приложений

Примеры заданий на лабораторные работы

№1 «Линейные программы»

- Напишите программу для расчета по двум формулам. Предварительно подготовьте тестовые примеры для второй формулы с помощью калькулятора (результаты вычисления по обеим формулам должны совпадать). Отсутствующие в языке функции выразите через имеющиеся.

$$z_1 = 2\sin^2(3\pi - 2\alpha)\cos^2(5\pi + 2\alpha)$$

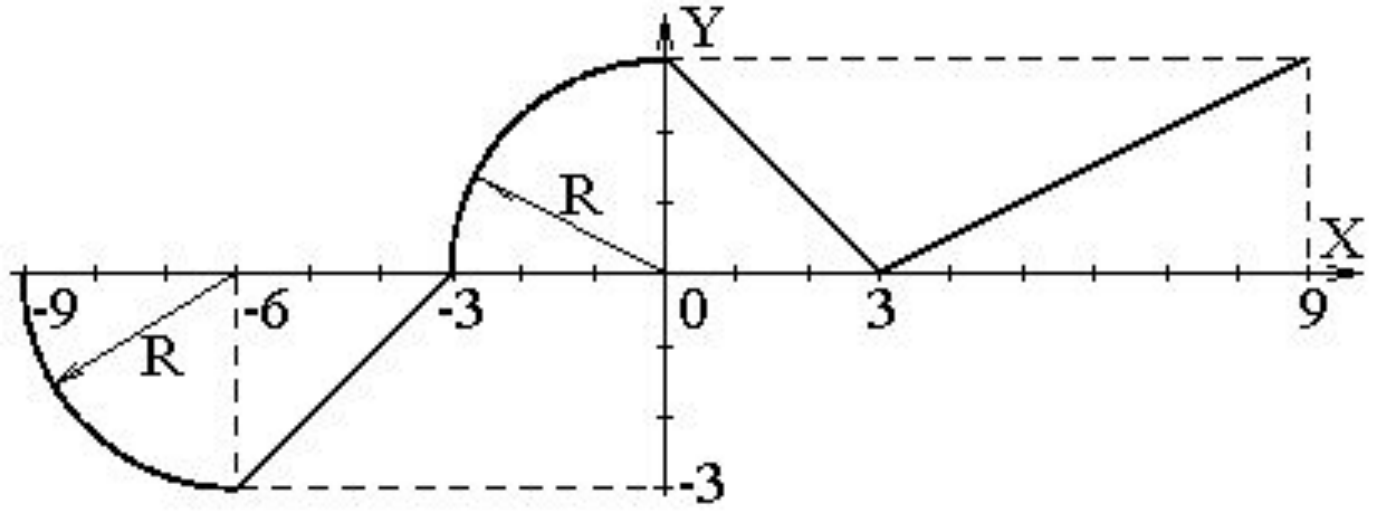
$$z_2 = \frac{1}{4} - \frac{1}{4}\sin(\frac{5}{2}\pi - 8\alpha)$$

Примеры заданий на лабораторные работы

№2 «Разветвляющиеся вычислительные процессы»

- Задание 1

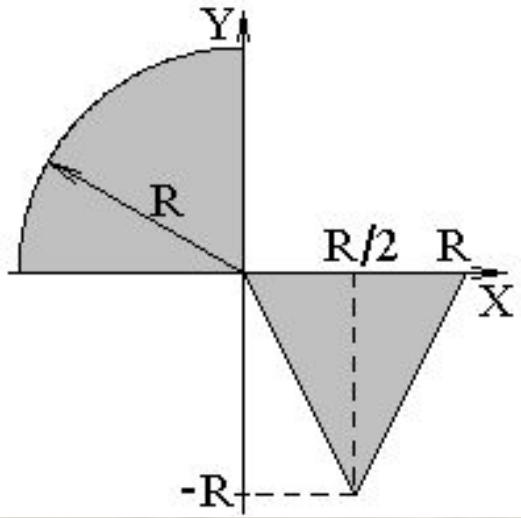
Написать программу, которая по введенному значению аргумента вычисляет значение функции, заданной в виде графика. Параметр *R* вводится с клавиатуры.

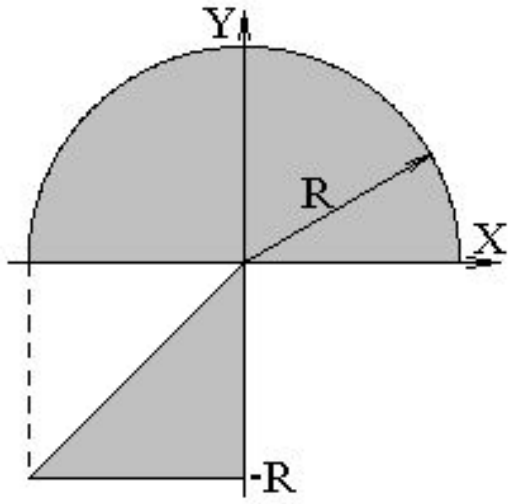


Примеры заданий на лабораторные работы

- Задание 2

Написать программу, которая определяет, попадает ли точка с заданными координатами в область, закрашенную на рисунке серым цветом. Результат работы программы вывести в виде текстового сообщения.





Примеры заданий на лабораторные работы

№3 «Организация циклов»

- Задание 3

Вычислить и вывести на экран в виде таблицы значения функции, заданной с помощью ряда Тейлора, на интервале от *x*нач до *x*кон с шагом *dx* с точностью *ε*. Таблицу снабдить заголовком и шапкой. Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.

$$e^{-x} = \sum_{n=0}^{\infty} \frac{(-1)^n x^n}{n!} = 1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \frac{x^4}{4!} - \ldots \qquad |x| < \infty$$

$$\ln(x+1) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{n+1}}{n+1} = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} - \ldots \qquad -1 < x \le 1$$

Примеры заданий на лабораторные работы

№4 «Одномерные массивы»

- В одномерном массиве, состоящем из *n* вещественных элементов, вычислить:
- Сумму отрицательных элементов массива.
- Произведение элементов массива, расположенных между максимальным и минимальным элементами.
- Упорядочить элементы массива по возрастанию.

№5 «Двумерные массивы и подпрограммы»

Дана целочисленная прямоугольная матрица. Определить:

1. Количество строк, не содержащих ни одного нулевого элемента (оформить в виде функции).
2. Максимальное из чисел, встречающихся в заданной матрице более одного раза (оформить в виде процедуры).

Примеры тестовых вопросов

- Выберите допустимые константы.
  1. 259
  2. 0,01
  3. 'ф'
  4. 1.2*e1
  5. $EEE
  6. '\\\'
  7. $FFH0
  8. ^C
  9. 'вася''

- Выберите все правильные ответы.

В переменной типа byte можно хранить число

1. 13;
2. 213;
3. –13;
4. –213;
5. 1213.

Примеры тестовых вопросов

- Чему равно числовое значение приведенного ниже выражения при e = 4, a = 3?

e + sqrt(e) * 1e1 / 2 * a

Чему равно значение выражения

(a and not b or c),

где a, b и c — величины типа boolean, имеющие значения true, false и false соответственно?

- Чему равно числовое значение приведенного ниже выражения при e = 4, a = 3?

e+Math.Sqrt(e)*1e1/2*a

Чему равно значение выражения

(! (a && b) || c),

где a, b и c — величины типа bool, имеющие значения true, false и false соответственно?

Примеры тестовых вопросов

- Какие выражения не содержат синтаксических ошибок?

1. sin(abs(0.6e3 * y_t))
2. a div b / c * mod
3. $EF01 * 1.34E–02 /_i7_17
4. 1_2i – exp(y) / 2 * t
5. cos(3 * x) + 0,2

- Какие выражения не содержат синтаксических ошибок?

1. -0.18*a/r-0.2*t
2. (-0.18)*a)/(r-0.2*t)
3. Math.Cos/^2*x+0,2
4. (-0.18)*a)\\(r-0.2*t))

- Выберите правильные ответы. Можно ли присвоить:

1. целую переменную — вещественной переменной;
2. вещественную переменную — целой переменной;
3. символьную переменную — целой переменной;
4. целую переменную — булевской переменной;
5. целой переменной — целую константу.

Примеры тестовых вопросов

Что из перечисленных ниже вариантов будет выведено на экран в результате работы программы?

```
var k : byte; x :real;
begin
    k := 2; x := 1;
    write ('*',   '*', x:3:1, k:k,
    '* *');
end.
```

1. **1.0 2* *
2. * * 1.0 2**
3. ** 1.02* *
4. **1.02**
5. ** 1.0 2* *

Чему будет равно значение переменной A после выполнения фрагмента программы, если B = 3?

```
A := 2; if B <= 4 then
C := 1 else C := 0; A := C;
```

Примеры тестовых вопросов

Чему будет равно значение переменной m после выполнения программы?

```
var k, m : word;
begin
   m := 1;
   for k := 3 downto 1 do;
   inc(m);
end;
```

Чему будет равно значение переменной m после выполнения фрагмента программы?

```
{
  int k;
  int m;
  k = 0; m = 0;
  while (k < 3)
    ++k;
    ++m;
}
```

Примеры тестовых вопросов

Что будет выведено на экран в результате работы программы?

```
const    a : array[1 .. 8] of char = 'abcrcaab';
var i : integer; k : string;
begin
  k := '';
  for i := 8 downto 2 do
    if a[i] < 'c' then k := k + a[i];
  write(k)
end.
```

Что будет выведено на экран в результате выполнения фрагмента программы?

```
{
char[] a = {'a', 'b', 'c', 'r', 'c', 'a', 'a', 'b'};
char k = 'b';
for (int i = 0; i < 8; i++)
   if (a[i] > k) k = a[i];
Console.Write(k);
}
```

Примеры тестовых вопросов

Заголовок подпрограммы имеет вид:

Procedure P(a : integer; var b, c : word; d : real);

Укажите правильные варианты вызова подпрограммы, если переменные в вызывающей программе описаны так:

var a : integer; b, k : word; e : integer; d, x : real;

1. P(a, b, k, d);
2. P(a, b, e, d);
3. P(k, b, k, x);
4. P(a + e div 2, b, k, d – x/2);
5. P(a, d, k, b);

Примеры тестовых вопросов

Какие из следующих утверждений являются НЕПРАВИЛЬНЫМИ?

```
class Base{};
class Derived: protected Base{};
```

1 Base является закрытым потомком Derived.
2 Base является суперклассом для Derived.
3 Derived является дочерним классом Base.
4 Derived является закрытым потомком Base.

Операция static_cast используется для преобразования типа между:
1 целыми типами
2 объектами, входящими в разные иерархии
3 целыми и перечисляемыми типами
4 целыми и вещественными типами

Пример экзаменационных требований для профильных специальностей

- Время написания — 80 минут. Баллов – 20.
- На экзамене требуется:

решить задачу (10 баллов)

пройти 3 теста: по основам синтаксиса, по операторам и по ООП (10 баллов).

Можно пользоваться литературой, но не компьютерами.