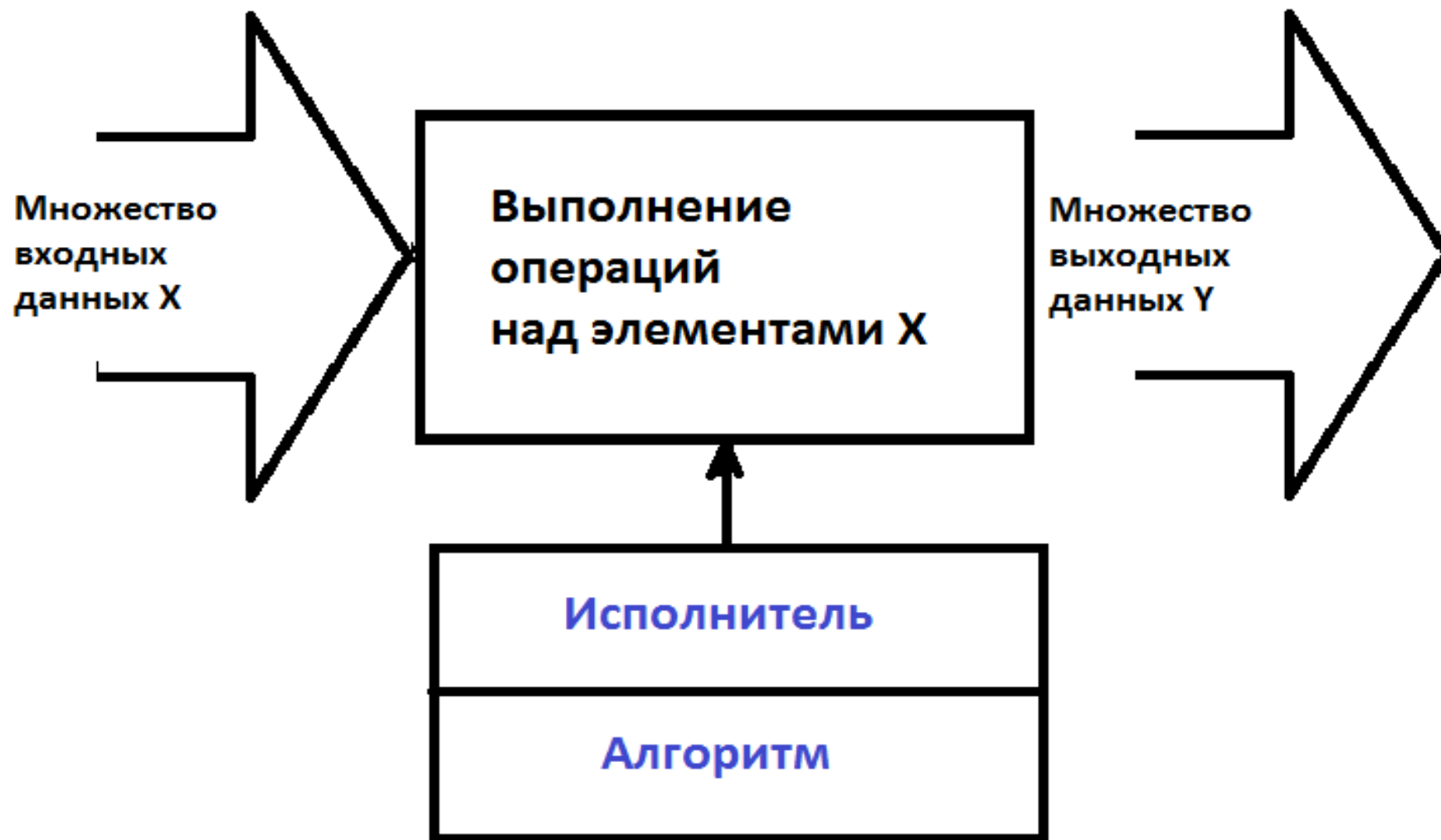


В.В. Подбельский

Иллюстрации к курсу лекций по дисциплине «Программирование на C#»

О языках программирования и
дисциплине «Программирование на C#»

Обработка Данных



Сравнение Языков Программирования по Парадигмам

- Императивная
- Объектно-ориентированная
- Функциональная
- Рефлексивная
- Обобщённая
- Логическая
- Декларативная
- Распределенная

СРАВНЕНИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ:

https://en.wikipedia.org/wiki/Comparison_of_programming_languages (англ.)

https://ru.m.wikipedia.org/wiki/Сравнение_языков_программирования (рус.)

Краткая Хронология Языков Программирования

Процедурные языки							
1954	1959	1960	1964	1964	1970	1972	1978-1983
FORTRAN Джон Бэкус	COBOL Грейс Хоппер	ALGOL IFIP	BASIC Томас Курц, Джон Кемени	PL/1 IBM	Pascal Никлаус Вирт	C Денис Ритчи	Ada Honeywell

Объектно-ориентированные языки			
1967	1969	1980	1985
Simula Кристен Ньюгорд, Оле-Йохан Даль	Smalltalk Алан Кэй	Smalltalk-80 Xerox PARK	C++ Бьерн Страуструп

«Мультипарадигменные» языки					
1987	1990	1994	1994	1995	2001
Perl Ларри Уолл	Python Гвидо ван Россум	PHP Расмус Лердорф	Java Sun Microsystems	Ruby Юкихиро Мацумото	C# Андерс Хейлсберг

Процедурные Языки

- 1954 – FORTRAN (Джон Бэкус)
- 1959 – COBOL (Грейс Хоппер)
- 1960 – ALGOL (IFIP)
- 1963 – BASIC (Джон Кемени и Томас Куртц)
- 1964 – PL/1 (IBM)
- 1970 – Pascal (Никлаус Вирт)
- 1972 – C (Денис Ритчи)
- 1978-1983 – Ada (Honeywell)

Объектно-Ориентированные Языки

- 1967 – Simula, Кристен Ньюгорд и Оле-Йохан Даль
- 1969 – Smalltalk, Алан Кэй;
- 1980 - Smalltalk-80, Xerox PARC
- 1985 – C++, Бьерн Страуструп

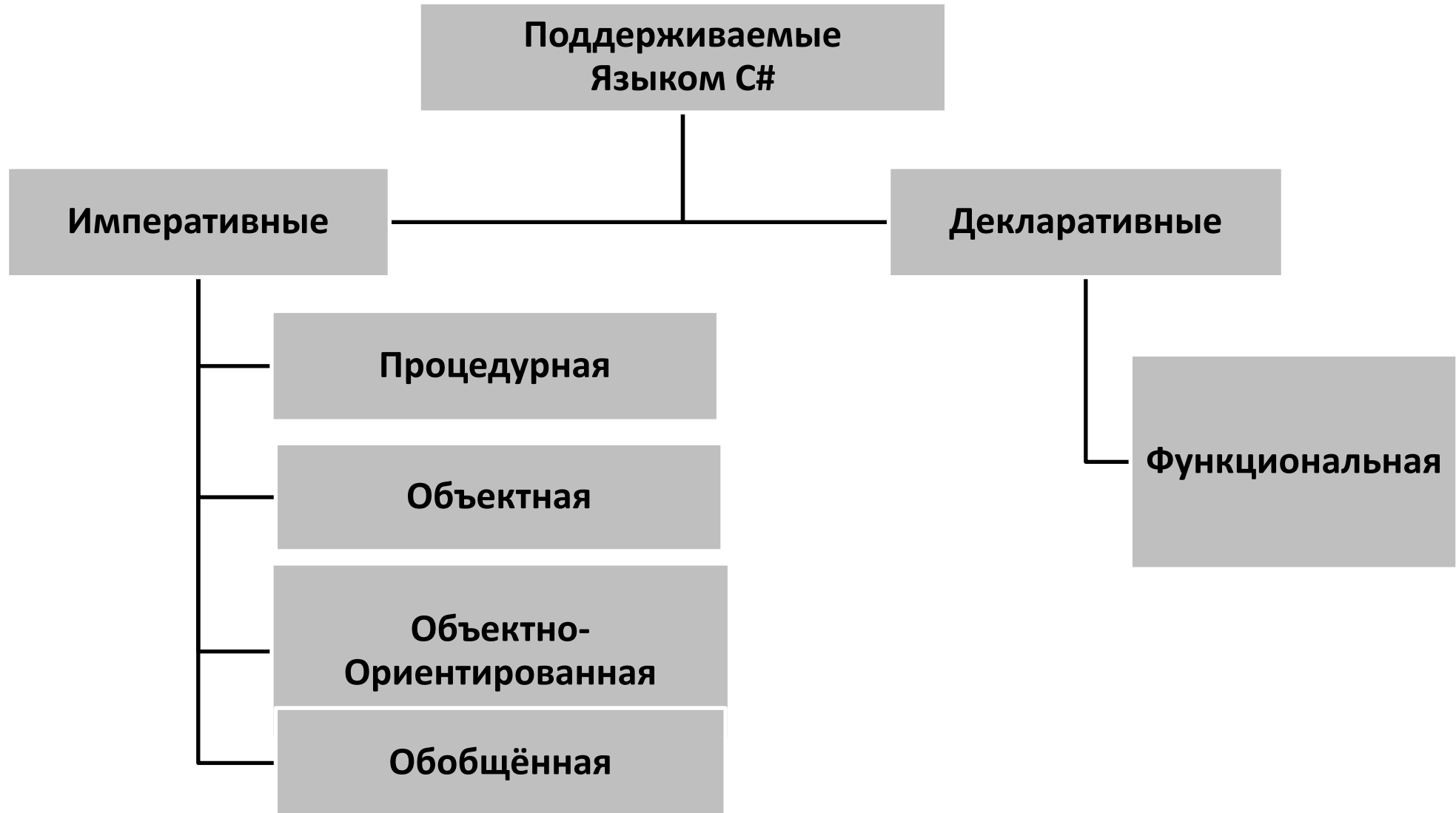
Декларативные Языки

- 1958 – Лисп (LISP), Джон Маккарти (язык обработки списков)
- 1972 - Prolog (Пролог – язык логического программирования)
- 1974 - SQL, Язык управления реляционными базами данных (Oracle).
- 1990 - Haskell – (функциональный язык, назван в честь математика Хаскелла Карри)
- 2008 - F#, Дон Сайм (Microsoft Research)

Мультипарадигмальные Языки

- 1987 – Perl, Ларри Уолл
- 1990 – Python, Гвидо ван Россум
- 1994 – PHP, Рasmus Лердорф
- 1994 – Java, Sun Microsystems
- 1995 – Ruby, Юкихиро Мацумото
- 2001 – C#, Microsoft, Андерс Хейлсберг

Парадигмы Программирования C#



Рубрики Классификации Языков

- Семантика
- Класс языка
- Появился
- Автор
- Выпуск
- Система типов
- Основные реализации
- Диалекты
- Испытал влияние
- На какие языки повлиял?
- Стандартизован ли?

Признаки Классификации Языков Программирования

- Парадигмы
- Типизация
- Компилятор / интерпретатор
- Управление памятью
- Управление потоком вычислений
- Типы и структуры данных
- Объектно-ориентированные возможности
- Функциональные возможности
- Разное
- Стандартизация

СРАВНЕНИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ:

https://en.wikipedia.org/wiki/Comparison_of_programming_languages (англ.)

https://ru.m.wikipedia.org/wiki/Сравнение_языков_программирования (рус.)

Поддержка Парадигм Языком C#

- | | | | |
|----------------------------|---------|------------------|---------|
| • Императивная | (+) | • Логическая | (-) |
| • Объектно-Ориентированная | (+) | • Доказательная | (-) |
| • Рефлексивная | (-/+) | • Декларативная | (-/+) |
| • Функциональная | (+/-) | • Распределённая | (-/+) |
| • Обобщённая | (+) | | |

(+)	указанная возможность присутствует
(-)	указанная возможность отсутствует
(+ / -)	возможность поддерживается не полностью
(- / +)	возможность поддерживается очень ограничено

Типизация

Статическая типизация	(+)	Вывод типов переменных из инициализатора	(+)
Динамическая типизация	(+)	Вывод типов переменных из использования	(-)
Явная типизация	(+)	Вывод типов-аргументов при вызове метода	(+)
Неявная типизация	(-/ +)	Вывод сигнатуры для локальных функций	(-)
Явное приведение типов	(+)	Параметрический полиморфизм	(+)
Неявное приведение типов без потери (точности)	(+)	Параметрический полиморфизм с ковариантностью	(+/-)
Неявное приведение типов с потерей данных	(+)	Параметрический полиморфизм высших порядков	(-)
Неявное приведение типов в неоднозначных ситуациях	(+)	Информация о типах в runtime	(+)
Алиасы типов	(+)	Информация о типах-параметрах в runtime	(+)

Возможности Разработки на C#

1. Настольные приложения (**Windows Forms/WPF**);
2. Мобильная разработка на **Xamarin** для iOS, Android;
3. Web-приложения с использованием **ASP.NET Core**;
4. Сервисы (**WCF**) и распределённые приложения;
5. Компьютерные 2D/3D игры с использованием движка **Unity**.

Механизм **ADO .NET** (ActiveX Data Objects .NET) для работы с базами данных.

- ADO.NET Entity Framework;
- LINQ (Language Integrated Query).

Microsoft Azure for Education (ex. Imagine, ex. DreamSpark)

1. Для получения доступа к личному кабинету необходимо использовать корпоративную почту: ...@edu.hse.ru.
2. Доступ в личный кабинет осуществляется по ссылке:
<https://portal.azure.com/#home>.

Полная информация о получении доступа к Microsoft Azure for Education: <https://it.hse.ru/dream>

План Преподавания Дисциплины в 2021-2022 году

Распределение аудиторных часов по модулям:

Модуль 1: в неделю: лек. 2, практ. 2.

Экзамен.

Модуль 2: в неделю: лек. 2, практ. 2.

Экзамен

Модуль 3: в неделю: лек. 2, практ. 2.

КДЗ. Экзамен

Итог: 200 часов аудиторных занятий.

Самостоятельная работа

Порядок Формирования Оценок

Условные обозначения:

- **НОi** – накопленная оценка i -го модуля;
- **СРi_j** – оценка самостоятельной работы j в i -м модуле (контекст / пиргрейд / тест);
- **ЭКi** – оценка за экзаменационный контекст (первая часть экзамена) i -го модуля;
- **ЭТi** – оценка за экзаменационный тест (вторая часть экзамена) i -го модуля;
- **ЭОi** – экзаменационная оценка i -го модуля (является неблокирующей в первом модуле и блокирующей во 2 и 3 модуле);
- **ИОi** – итоговая оценка i -го модуля;

Порядок Формирования Оценок 1 Модуля

ИО1 = Округление($0,5 * ЭО1 + 0,5 * НО1$).

НО1 = $(CP1_1 + CP1_2 + CP1_3 + CP1_4 + CP1_5 + CP1_6) / 6$.

Если экзамен состоит из двух частей (контест и тест):

ЭО1 = $0,5 * ЭК1 + 0,5 * ЭТ1$, если $\min(ЭК1; ЭТ1) \geq 3,5$;

иначе **ЭО1** = $\min(ЭК1; ЭТ1)$.

Обозначения:

ЭК<і> - экзаменационный контест <і>-го модуля;

ЭТ<і> - экзаменационный тест <і>-го модуля.

Порядок Формирования Оценок 2 Модуля

ИО2 = Округление($0,5 \cdot \text{ЭО2} + 0,5 \cdot \text{НО2}$).

НО2 = $(\text{CP2_1} + \text{CP2_2} + \text{CP2_3} + \text{CP2_4} + \text{CP2_5} + \text{CP2_6}) / 6$.

Экзамен состоит из двух частей (контест и тест):

ЭО2 = $0,5 \cdot \text{ЭК2} + 0,5 \cdot \text{ЭТ2}$, если $\min(\text{ЭК2}; \text{ЭТ2}) \geq 3,5$;

иначе **ЭО2** = $\min(\text{ЭК2}; \text{ЭТ2})$.

Обозначения:

ЭК<і> - экзаменационный контест <і>-го модуля;

ЭТ<і> - экзаменационный тест <і>-го модуля.

Порядок Формирования Оценок 3 Модуля

ИОЗ = Округление($0,5 * ЭОЗ + 0,5 * НОЗ$).

НОЗ = $(CP3_1 + CP3_2 + CP3_3 + CP3_4 + CP3_5 + CP3_6 + CP3_7 + CP3_8 + CP3_9) / 9$.

Экзамен состоит из двух частей (контест и тест):

ЭОЗ = $0,5 * ЭКЗ + 0,5 * ЭТЗ$, если $\min(ЭКЗ; ЭТЗ) \geq 3,5$;
иначе **ЭОЗ** = $\min(ЭКЗ; ЭТЗ)$.

Обозначения:

ЭК<і> - экзаменационный контест <і>-го модуля;

ЭТ<і> - экзаменационный тест <і>-го модуля.

Критерии Выставления Оценки за Программу

“НЕУДОВЛЕТВОРИТЕЛЬНО”:

1 балл:

1. Разработка программы не завершена.
2. Программа имеет синтаксические ошибки (не компилируется).

2 балла:

1. Программа не решает основную задачу или не соответствует спецификации.
2. В программе обнаруживаются не обработанные исключения при решении основных и второстепенных подзадач.

3 балла:

1. Программа не решает основную задачу при некоторых вариантах исходных данных.
2. Программа завершается аварийно при некоторых вариантах исходных данных.

Критерии Выставления Оценки за Программу

“УДОВЛЕТВОРИТЕЛЬНО”:

4 балла:

1. Программа решает основную задачу, но имеет отклонения от спецификации.

5 баллов:

1. Программа соответствует критериям получения оценки 4 балла.
2. Программа соответствует отдельным дополнительным критериям.

Критерии Выставления Оценки за Программу

“ХОРОШО”:

6 баллов:

1. Программа решает поставленную задачу и соответствует спецификации. Отклонения от спецификации допущены при реализации второстепенных подзадач.
2. Исходный текст документирован.

7 баллов:

1. Программа соответствует критериям получения оценки 6 баллов.
2. Программа в целом соответствует дополнительным критериям.

Критерии Выставления Оценки за Программу

“ОТЛИЧНО”:

8 баллов:

1. Программа решает все поставленные задачи и полностью соответствует спецификации.
2. Студент в комментариях обосновал принятые конструктивные решения.
3. Исходный текст документирован. Присутствуют сведения о назначении используемых переменных, параметров, методов, классов, объектов.
4. Программа остается работоспособной при вводе неверных исходных данных.
5. Предусмотрено повторное решение задачи без повторного запуска программы.
6. Программа реализована по модульному принципу и хорошо декомпозирована.

Критерии Выставления Оценки за Программу

“ОТЛИЧНО”:

9 баллов:

1. Программа соответствует критериям получения оценки 8 баллов.
2. Программа соответствует некоторым дополнительным критериям.

10 баллов:

1. Программа соответствует критериям получения оценки 8 баллов.
2. Программа полностью соответствует всем дополнительным критериям.
3. Студент отразил в комментариях возможность альтернативных вариантов решения задачи.

Дополнительные Критерии Оценивания

1. Исходный текст программы структурирован.
2. Программа имеет средства изменения размерности и формы представления данных без изменения исходного текста или при минимальных изменениях (динамическое выделение памяти, именованные константы, и т.п.).
3. Программа реализована по модульному принципу и соответствующим образом декомпозирована.
4. В коде программы обоснованно использованы конструкции, изученные студентом самостоятельно, и их применение не противоречит основным требованиям, предъявляемым к решаемой задаче.
5. Предусмотрено самодокументирование программы.

Где искать материалы?

Команда MS Teams “Программирование на C# (2021-2022)”:

— <http://bit.do/cshse2021>

Работа в компьютерном классе

- В начале сеанса работы в классе студент должен на диске **D** создать папку и назвать ее своей фамилией.
- В папке следует сохранять результаты работы на практическом занятии.



Внимание: После выключения компьютера, все папки с рабочего стола и диска C:\ автоматически удаляются!

План Проверочных Работ (Модуль 1)

- Контест: 07.09-13.09
- Peer Review: 14.09-20.09
- Контест: 21.09-27.09
- Тест: 28.09-04.10
- Контест: 05.10-11.10
- Peer Review: 12.10-17.10

Экзамен - 18-24 октября