



## I. Пояснительная записка

Авторы программы: программа дисциплины разработана коллективом кафедры информационных систем и технологий НФ ГУ ВШЭ, ответственный исполнитель ст. преп. Маслова Е.А.

*Требования к студентам:*

для освоения дисциплины студент должен владеть современными методами и средствами информатики, быть знакомым с принципами алгоритмизации.

*Аннотация:*

Дисциплина рассчитана на студентов, специализирующихся в области бизнес-информатики и прикладной математики.

Цели дисциплины: овладение студентами принципами функционального программирования, формирование у них понимания места функционального подхода в процессе разработки программного обеспечения с интеллектуальными возможностями в современных условиях, круга решаемых задач, а также ознакомление с используемыми инструментальными средствами.

Дисциплина рассчитана на 21 час лекционных и 28 часов практических занятий.

*Учебная задача дисциплины:*

В результате изучения дисциплины студент должен:

- **знать** принципы функционального программирования, их связь с формальными методами алгоритмизации и методами искусственного интеллекта, особенности синтаксиса и семантики распространенных языков программирования функционального типа;
- **уметь применять** полученные теоретические знания к решению вопросов проектирования, разработки и тестирования программного обеспечения для решения сложных информационных проблем, с которыми сталкиваются конкретные предприятия;
- **иметь представление** о возможностях современных средств разработки на языках программирования функционального типа, методах интеграции с другими применяемыми на практике информационными технологиями;
- **обладать навыками** представления различных задач и структур данных в терминах функционального подхода, а также оценки адекватности применения этого подхода для решения конкретных задач пользователя.

## II. Тематический план учебной дисциплины

	Наименование тем	Всего часов	Аудиторные часы		Самостоятельная работа
			Лекции	Практические занятия	
1.	Исторический обзор средств программирования, разъяснение причин необходимости использования функционального подхода для создания интеллектуальных систем	7	1	0	6
2.	Основные синтаксические конструкции языка функционального типа на примере языка Scheme.	7	1	2	4
3.	Композитные типы данных в функциональных языках программирования	6	1	2	3
4.	Принципы реализации базовых конструкций программирования в языках функционального типа	5	1	2	2
5.	Организация программных проектов на языках функционального типа	5	1	2	2
6.	Формальные основы функционального программирования	10	2	0	8
7.	Обработка сложных структур данных	8	2	4	2
8.	Общие принципы проектирования функциональных приложений	6	2	2	2
9.	Особые приемы программирования интеллектуальных систем	13	2	4	7
10.	Применение концепции иерархии виртуальных машин	9	2	0	7
11.	Объектно-ориентированные возможности функциональных языков	6	2	0	4
12.	Графические возможности современных функциональных языков	5	1	2	2

	Наименование тем	Всего часов	Аудиторные часы		Самостоятельная работа
			Лекции	Практические занятия	
13.	Обзор современных сред программирования для языков функционального типа	3	1	0	2
14.	Функциональное программирование для реализации Internet-приложений	9	1	4	4
15.	Технологии интеграции функциональных программ с другими программными компонентами	9	1	4	4
<b>Итого</b>		<b>108</b>	<b>21</b>	<b>28</b>	<b>59</b>

### III. Формы рубежного и итогового контролей

Контроль знаний студентов включает формы текущего и итогового контроля. Текущий контроль осуществляется в течение двух модулей. По курсу предусмотрены текущий контроль знаний и работы студентов на практических занятиях, две контрольные работы (2 x 120 мин.) и самостоятельная разработка программных проектов на дому (эссе). Каждая форма текущего контроля оценивается 10-балльной оценкой, которая выставляется в рабочую ведомость преподавателя. По результатам текущего контроля организуются индивидуальные консультации в рамках второй половины рабочего дня преподавателя. Форма итогового контроля – письменный дифференцированный зачет по окончании двух модулей учебного курса, который оценивается по 10-балльной шкале. Продолжительность зачета 120 мин.

Для получения *результатирующей оценки О* итогового контроля используются следующие весовые множители:

- 0,2 – для оценки **Оработ** за работу студентов на практических занятиях,
- 0,15 – для оценки **Оконтр1.**, за контрольную работу № 1,
- 0,15 – для оценки **Оконтр2.**, за контрольную работу № 2,
- 0,2 – для оценки **Оэссе.**, за самостоятельная разработка программы на дому,
- 0,3 – для оценки **Описьм.зач.**, за письменный зачет.

Для получения *результатирующей оценки О* по 10-балльной шкале вычисляется величина

$$O = 0,2 \times \text{Оработ} + 0,15 \times \text{Оконтр.1} + 0,15 \times \text{Оконтр.2} + 0,2 \times \text{Оэссе} + 0,3 \times \text{Описьм.зач.}$$

Полученный после округления этой величины до целого значения результат и *выставляется* как *результатирующая оценка по 10-балльной шкале* по учебной дисциплине «Функциональное

программирование» в экзаменационную ведомость и зачетную книжку студента. В экзаменационную ведомость и зачетную книжку студента выставляется также и результат зачета по данной дисциплине **в форме «зачтено» «не зачтено»**, получаемый из оценки по десятибалльной шкале в соответствии со следующей таблицей соответствия (см. Приложение № 2 к приказу Ректора ГУ-ВШЭ № 1002 от 17.06.2002).

**Таблица соответствия оценок по десятибалльной системе и результата зачета.**

По десятибалльной шкале	Результат
1 – неудовлетворительно 2 – очень плохо 3 – плохо	<b>«не зачтено»</b>
4 – удовлетворительно 5 – весьма удовлетворительно	<b>«зачтено»</b>
6 – хорошо 7 – очень хорошо	<b>«зачтено»</b>
8 – почти отлично 9 – отлично 10 – блестяще	<b>«зачтено»</b>

#### **IV. Содержание программы**

##### **Тема 1. Исторический обзор средств программирования, разъяснение причин необходимости использования функционального подхода для создания интеллектуальных систем**

Тенденции развития аппаратного и программного обеспечения. Эволюция требований пользователей. Абстракция и унификация программных компонент. Императивные и декларативные языки программирования. Недостатки языков программирования процедурного типа. Особенности функционального программирования. Обзор проблематики и областей применения искусственного интеллекта.

##### *Основная литература*

1. Хювенен Э., Сеппянен Й., Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.
2. Хендерсон П. Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.

##### *Дополнительная литература*

3. Хювенен Э., Сеппянен Й., Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
4. Alan J. Perlis Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).
5. Джексон П. Введение в экспертные системы. – М.: Вильямс, 2001. – 624с.

## **Тема 2. Основные синтаксические конструкции языка функционального типа на примере языка Scheme.**

Простые типы данных. Определение переменных. S-выражения. Формы, функции и процедуры. Передача параметров. Функционал apply

### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi.* How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хювенен Э., Сеппянен Й.,* Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.

### *Дополнительная литература*

3. *Dorai Sitaram* Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
4. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

## **Тема 3. Композитные типы данных в функциональных языках программирования**

Общие принципы организации системы типов в процедурных языках на примере Scheme.

Особенности числовых типов данных в языке программирования Scheme. Строки. Векторы. Точечные пары. Принципы преобразования типов. Равенство значений. Структуры. Списки. Внутреннее представление списков. Функции работы со списками. Ассоциативные списки.

### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi.* How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хювенен Э., Сеппянен Й.,* Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.
3. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.

### *Дополнительная литература*

4. *Dorai Sitaram* Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
5. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

## **Тема 4. Принципы реализации базовых конструкций программирования в языках функционального типа**

Условные выражения. Связывание переменных со значениями. Функции ввода-вывода.

Применение средств макроразстановки в языке Scheme и Lisp. Интерфейс к операционной системе. Композиция функций. Циклы. Повторение через итерацию и рекурсию.

### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.
3. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.  
*Дополнительная литература*
4. *Dorai Sitaram* Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
5. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

### **Тема 5. Организация программных проектов на языках функционального типа**

Состав и возможности программной системы DrScheme. Структура программного проекта. Возможности по созданию и отладке программ. Интерактивная система помощи. Назначение модулей и пакетов в DrScheme. Настройка параметров программной системы DrScheme.

#### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.
3. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.  
*Дополнительная литература*
4. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
5. *Dorai Sitaram* Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
6. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

### **Тема 6. Формальные основы функционального программирования**

Символьные данные. Элементарные селекторы и конструкторы. Рекурсивные функции. Функции высших порядков. Лямбда-исчисление. Машинная архитектура для функционального программирования.

#### *Основная литература*

1. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.  
*Дополнительная литература*
2. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge,

Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

### **Тема 7. Обработка сложных структур данных**

Важность использования эффективных структур данных при решении сложных интеллектуальных задач. Параллельное ветвление рекурсии. Взаимная рекурсия. Рекурсия высоких порядков. Представление стеков, очередей, последовательностей, деревьев. Взаимно-референтные структуры данных. Мутабельные структуры данных.

#### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.
3. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.

#### *Дополнительная литература*

4. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
5. *Dorai Sitaram* Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
6. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

### **Тема 8. Общие принципы проектирования функциональных приложений**

Абстракция и декомпозиция. Разработка функциональных абстракций на примерах. Контракты на интерфейсы. Абстракция данных. Альтернативные представления абстрактных типов данных. Программирование, управляемое данными и аддитивность. Методы обработки ошибок и исключительных ситуаций.

#### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.1: Введение в язык Лисп и функциональное программирование. – М.: Мир, 1990. – 447с.
3. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.

#### *Дополнительная литература*

4. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
5. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).



## **Тема 9. Особые приемы программирования интеллектуальных систем**

Генерирующая рекурсия. Алгоритмы с возвратами. Аккумуляция данных. Нелокальные переходы. Недетерминированные вычисления. Отложенные вычисления.

### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.

### *Дополнительная литература*

3. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
4. *Dorai Sitaram* Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
5. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).
6. *Джексон П.* Введение в экспертные системы. – М.: Вильямс, 2001. – 624с.

## **Тема 10. Применение концепции иерархии виртуальных машин**

Реализация металингвистической абстракции средствами функциональных языков. Метациркулярный вычислитель на языке Scheme. Представление выражений. Программы перевода. Симулятор регистровой машины. Структура и функции ассемблера регистровой машины. Компилятор.

### *Основная литература*

1. *Alan J. Perlis* Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).
2. *Хендерсон П.* Функциональное программирование. Применение и реализация. – М.: Мир, 1983. – 349с.

### *Дополнительная литература*

3. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
4. *Джексон П.* Введение в экспертные системы. – М.: Вильямс, 2001. – 624с.

## **Тема 11. Объектно-ориентированные возможности функциональных языков**

Объект, класс и метакласс. Наследование. Порядок наследования в иерархии классов. Полиморфизм. Инкапсуляция. Взаимодействие на основе передачи сообщений. Различные возможности объектно-ориентированных расширений на примере объектно-ориентированного расширения Scheme и Flavors. Применимость объектного программирования.

### *Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)

### *Дополнительная литература*

2. Хювенен Э., Сеппянен Й., Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
3. Dorai Sitaram Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).
4. Джексон П. Введение в экспертные системы. – М.: Вильямс, 2001. – 624с.

### **Тема 12. Графические возможности современных функциональных языков**

Базовые примитивы системы DrScheme. Алгоритмы управления отображением графических объектов. Применение приемов программирования для реализации графических приложений. Средства создания графического интерфейса пользователя в системе DrScheme.

#### *Основная литература*

1. Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. Dorai Sitaram Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).

### *Дополнительная литература*

3. Хювенен Э., Сеппянен Й., Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
4. Alan J. Perlis Structure and Interpretation of Computer Programs. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

### **Тема 13. Обзор современных сред программирования для языков функционального типа**

Диалекты Scheme (DrScheme, MIT Scheme и др.). Первичная среда Common Lisp. Среда Allegro Common Lisp. Среда Zlisp. Направления развития функциональных языков программирования. Возможности языков смешанного типа на примере Tcl/Тк.

#### *Основная литература*

1. Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. Dorai Sitaram Teach yourself Scheme in Fixnum Days. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).

### *Дополнительная литература*

3. Хювенен Э., Сеппянен Й., Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.

### **Тема 14. Функциональное программирование для реализации Internet-приложений**

Архитектура современных Internet-приложений. Стандарт CGI. Методы создания CGI-скриптов средствами DrScheme. Утилитные функции для создания CGI-скриптов.

#### *Основная литература*

1. Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)

2. *Dorai Sitaram Teach yourself Scheme in Fixnum Days*. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).

*Дополнительная литература*

3. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.

**Тема 15. Технологии интеграции функциональных программ с другими программными компонентами**

Интерфейсы связи с базами данных на основе ODBC. Методы построения распределенных объектно-ориентированных систем. Технология CORBA. Реализация распределенных функциональных алгоритмов на основе технологии CORBA. Технология ActiveX. Методы интеграции программ на языке Scheme с использованием технологии ActiveX.

*Основная литература*

1. *Matthias Felleisen Robert Bruce Findler Matthew Flatt Shriram Krishnamurthi*. How to design programs. An introduction to Programming and Computing. The MIT Press Cambridge, Massachusetts London, England (электронный вариант: <http://www.htdp.org/2003-09-26/Book/>)
2. *Dorai Sitaram Teach yourself Scheme in Fixnum Days*. (электронный вариант: <http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html>).

*Дополнительная литература*

3. *Хювенен Э., Сеппянен Й.*, Мир Лиспа. В 2х. т. Т.2: Методы и системы программирования. – М.: Мир, 1990. – 319с.
4. *Alan J. Perlis Structure and Interpretation of Computer Programs*. The MIT Press Cambridge, Massachusetts London, England McGraw-Hill Book Company New York St. Louis San Francisco Montreal Toronto (электронный вариант: <http://mitpress.mit.edu/sicp/full-text/sicp/book/book.html>).

## **V. Тематика самостоятельных работ**

### **1. Числа, выражения, простые программы**

1. Разработать программу для вычисления площади треугольника, по известной длине стороны и его высоте.
2. Разработать программу для вычисления подоходного налога по ставке 15%, исходя из суммы дохода.
3. Разработать программу для определения прибыли кинотеатра, из расходов 20\$ с сеанса + 0.5\$ с каждого посетителя, полагая, что один билет стоит 5\$

### **2. Программы – это функции и определения переменных**

1. Разработать программу для определения количества посетителей кинотеатра при различной цене билета, по правилу: при цене в 5\$ - 120 человек, уменьшение цены на 0.1\$ приводит к увеличению посетителей на 15 человек
2. Разработать программу прибыли, приносимой кинотеатром при известной цене билета.
3. Разработать программу для вычисления высоту, на которую поднялась ракета в заданный момент времени по формуле  $\frac{gt^2}{2}$ .

### **3. Условные выражения и функции**

1. Разработать программу для определения является ли указанное число корнем следующих уравнений:
  - $4n + 2 = 62$
  - $2n^2 = 102$
  - $4n^2 + 6n + 2 = 462$

2. Разработать программу для нахождения величины наращенной суммы (S) по вкладу (P) при следующих условиях нахождения простой процентной ставки (i)
  - если  $P < 1000$ , то  $i = 4.5\%$
  - если  $P \geq 1000$  и  $P < 5000$ , то  $i = 4.5\%$
  - если  $P \geq 5000$ , то  $i = 5\%$
3. Разработать программу для нахождения количества корней квадратного уравнения по его заданным коэффициентам.

#### 4. Символьная информация

1. Разработать программу для «игры» по следующим правилам: первый игрок загадывает число, а второй пытается угадать его, программа должна вернуть результат сравнения.
2. Разработать игру аналогично 1, за исключением того что число, загадываемое первым игроком, представляет собой множество цифр, соответствующих разряду числа (начиная с младших)

#### 5 Составные типы данных: Структуры

1. Разработать структуру для описания военного самолёта с 4-мя основными характеристиками: модель, ускорения, максимальная скорость, дальность полёта. Разработать функцию `within-range`, которая принимает самолёт и расстояние от базы до цели, сообщая при этом сможет ли самолёт долететь до цели. Разработать функцию `reduce-range`, которая принимает самолёт и уменьшает дальность полёта до 80% от первоначальной
2. Разработать структуру для представления времени. Время представляется тремя числами: часы, минуты и секунды.
3. Разработать функцию `time->seconds`, которая принимает на вход структуру `time` из упражнения 2 и выдаёт время в секундах, начиная с полуночи. Разработать функцию `draw-and-clear-circle`, которая рисует окружность, ждёт некоторое время и стирает её.
4. Разработать структуру для представления трёхбуквенных слов.
5. Разработать функцию `reveal`, которая принимает 3 аргумента:
  - слово, которое мы отгадываем;
  - слово, которое показывает какая часть была открыта до сих пор, «закрытые» буквы отмечаются символом '\_';
  - угадываемая буква.

Полученное слово должно состоять из букв и символов '\_' и образуется по следующим правилам:

Если названная буква совпадает с буквой в загаданном слове, то она появляется в полученном слове.

Если угадываемая буква отсутствует, то возвращается слово, передаваемое в качестве второго параметра.

#### 6. Обработка смешанных данных

1. Разработать функцию `area`, которая принимает на вход окружность или квадрат и вычисляет площадь.
2. Разработать функцию `move-shape`, которая передвигает фигуру, по известному расстоянию сдвига и типу фигуры.

#### 7. Составные типы данных: Списки

1. Разработать тип данных – список из двух символов. Разработать функцию `contains-2-doll?`, которая принимает на вход список и определяет содержится ли в нём слово 'doll'.
2. Разработать функцию `contains?`, которая принимает на вход символ и список из символов, при этом возвращает `true`, если символ содержится в списке и `false` – иначе.
3. Разработать функцию `draw-circles`, которая принимает на вход координату точки `p` и список чисел. Каждое число в списке представляет собой радиус некоторой окружности. Функция должна нарисовать окружности вокруг точки `p`.

#### 8. Обработка списков

1. Разработать функцию `quadratic-roots`, решающую квадратное уравнение, функция принимает на вход коэффициенты `a`, `b`, и `c`.
2. Телефонная книга содержит имена и телефоны. Разработать структуру данных телефонная запись и телефонная книга, а также функции
  - `whose-number`, которая по номеру определяет имя;
  - `phone-number`, которая по имени выдаёт номер.

## VI. Методические рекомендации преподавателю

См., приложение по методике 10-балльной оценки.

## VII. Методические рекомендации студентам

При выполнении домашнего задания поощряется использование информации из профессиональных публикаций и разработок, из Интернет-источников, но с обязательной

ссылкой на адрес сайта, авторов использованных материалов, а также самостоятельное изучение и анализ различных программных реализаций языков функционального типа. Предпочтительной представляется работа, выполненная, например, в виде адаптации существующего программного решения на другом языке программирования с указанием изменений, выполненных в оригинальном алгоритме с оценкой характеристик полученного решения. Приветствуется работа с актуальными материалами из зарубежной профессиональной периодики.

## **VIII. Рекомендации по использованию информационных технологий**

В НФ ГУ ВШЭ студентам предоставляется возможность самостоятельной работы с электронными ресурсами информации, периодической литературой. В компьютерном классе (в здании на Б. Печерской) доступ on-line. Для изучения методов программирования используется свободно-распространяемое обеспечение — программная система DrScheme.

Преподаватель

Маслова Е.А.

## **Приложение: Методика формирования результирующей оценки**

1. Программа предусматривает в качестве форм контроля: оценку контрольной работы, оценку домашнего задания, зачет (экзамен). Каждая форма оценивается по 10-бальной шкале.

### Для контрольных работ:

высшая оценка в 9 баллов (10 баллов проставляется в исключительных случаях) проставляются при отличном выполнении заданий: полных (с детальными или многочисленными примерами и возможными обобщениями) ответах на вопросы, правильном решении задачи и четком и исчерпывающем ее представлении,

почти отличная оценка в 8 баллов проставляется при полностью правильных ответах и решении задач, но при отсутствии какого-либо из выше перечисленных отличительных признаков, как, например: детальных примеров или обобщений, четкого и исчерпывающего представления решаемой задачи,

оценка в 7 баллов проставляется при правильных ответах на вопросы и правильном решении задачи, но при отсутствии пояснений, примеров, обобщений, без представления алгоритма или последовательности решения задач,

оценка в 6 баллов проставляется при наличии отдельных неточностей в ответах на вопросы (включая грамматические ошибки) или неточностях в решении задачи не принципиального характера (описки и случайные ошибки арифметического характера),

оценка в 5 баллов проставляется в случаях, когда в ответах и в решении задач имеются неточности и ошибки, свидетельствующие о недостаточном понимании вопросов и требующие дополнительного обращения к тематическим материалам,

оценка в 4 балла проставляется при наличии серьезных ошибок и пробелов в знании по контролируемой тематике,

оценка в 3 балла проставляется при наличии лишь отдельных положительных моментов в ответах на вопросы и в решении задач, говорящих о потенциальной возможности в последующем более успешно выполнить задания; оценка в 3 балла, как правило, ведет к повторному написанию ответов на вопросы или решению дополнительной задачи,

оценка в 2 балла проставляется при полном отсутствии положительных моментов в ответах на вопросы и решении задач и, как правило, ведет к повторному написанию контрольной работы в целом,

оценка в 1 балл проставляется, когда неправильные ответы и решения, кроме того, сопровождаются какими-либо демонстративными проявлениями безграмотности или неэтичного отношения к изучаемой теме.

### При оценке выполнения домашнего задания (эссе):

10 баллов проставляется в исключительных случаях самостоятельно проведенной работы, которая может в дальнейшем использоваться в учебном процессе или в исследовательской работе студента,

8-9 баллов проставляется при самостоятельно разработанном или удачно адаптированном и отлично представленном исследовании по выбранной тематике,

6-7 баллов проставляется при своевременно выполненном и самостоятельно представленном результате продукте по выбранной тематике,

4-5 баллов проставляется при частичном, несамостоятельном участии в выполнении работ,

2-3 балла проставляется, когда студент не может самостоятельно представить работу; когда работа носит явные признаки заимствований (работу предлагается переделать),

1 балл проставляется при наличии каких-либо демонстративных проявлений

безграмотности и неэтичного отношения к работе.

На зачете (экзамене), представляющем собой письменный ответ на вопрос и решение задачи с последующим собеседованием, оценка проставляется следующим образом:

высшая оценка в 9 баллов (10 баллов проставляется в исключительных случаях) проставляется при отличном выполнении заданий: полных, с примерами и возможными обобщениями ответов на вопросы, при правильном решении задачи и детальном ее представлении,

почти отличная оценка в 8 баллов проставляется при полностью правильных ответах и решении задач, но при отсутствии какого-либо из выше перечисленных отличительных признаков, как, например: примеров и обобщений, детального представления решаемой задачи,

оценка в 7 баллов проставляется при правильных ответах на вопросы и правильном решении задачи, но при отсутствии пояснений, примеров, без представления алгоритма решения задач,

оценка в 6 баллов проставляется при наличии отдельных неточностей в ответах на вопросы или непринципиальных неточностях в решении задачи (описки и случайные ошибки арифметического характера),

оценка в 4-5 баллов проставляется в случаях, когда в ответах и в решении задачи имеются существенные неточности и ошибки, свидетельствующие о недостаточном понимании проблематики,

оценка в 2-3 балла проставляется при наличии лишь отдельных положительных моментов в ответах на вопросы и в решении задачи и ведет к повторному написанию ответов на вопросы или решению задачи,

оценка в 1 балл проставляется, когда неправильные ответы и решения, кроме того, сопровождаются какими-либо демонстративными проявлениями безграмотности или неэтичного отношения к учебному процессу.

**По результатам устного собеседования с преподавателем выполненной на экзамене работы возможны корректировки оценки в ту или иную сторону.**