

**Санкт-Петербургский филиал федерального государственного  
автономного образовательного учреждения высшего образования  
"Национальный исследовательский университет  
"Высшая школа экономики"**

Факультет Санкт-Петербургская школа  
физико-математических и компьютерных наук  
Департамент информатики

**Рабочая программа дисциплины  
Современные языки программирования**

для майнора «Основы программирования»  
уровень бакалавриат

Разработчик: Кузнецов Антон Михайлович, amkuznetsov@hse.ru

Утверждена Академическим руководителем образовательной программы

«31» августа 2018 г.

А.В. Омельченко \_\_\_\_\_

Санкт-Петербург, 2018

*Настоящая программа не может быть использована другими подразделениями университета и другими вузами без разрешения подразделения-разработчика программы.*

## 1. Область применения и нормативные ссылки

Настоящая программа учебной дисциплины устанавливает требования к образовательным результатам и результатам обучения студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих дисциплину «Современные языки программирования», учебных ассистентов и студентов направления 01.03.02 «Прикладная математика и информатика» подготовки бакалавра, обучающихся по бакалаврской программе «Прикладная математика и информатика» и изучающих дисциплину «Современные языки программирования».

Рабочая программа дисциплины разработана в соответствии с:

- Образовательным стандартом НИУ ВШЭ по направлению подготовки 01.03.02 «Прикладная математика и информатика» (уровень бакалавриата), утвержденным ученым советом Национального исследовательского университета «Высшая школа экономики», протокол от 03.03.2017 №02.
- Образовательной программой «Прикладная математика и информатика» направления подготовки 01.03.02 «Прикладная математика и информатика»;
- Объединенным учебным планом университета по образовательной программе «Прикладная математика и информатика», утвержденным в 2018 г.

## 2. Цели освоения дисциплины

Целями освоения дисциплины «Современные языки программирования» являются формирование у студентов теоретических знаний и практических навыков по основам методологии программирования.

Курс дает представление о современных языках программирования и их роли в промышленной разработке. На примере таких языков, как Java, Kotlin и Scala студенты научатся разрабатывать приложения для персональных компьютеров и мобильных устройств.

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины студент осваивает следующие компетенции:

Компетенция	Код по ФГОС	Уровень формирования компетенции	Дескрипторы – основные признаки освоения (показатели достижения результата)	Формы и методы обучения, способствующие формированию и развитию компетенции	Форма контроля уровня сформированности компетенции
Способность учиться, приобретать новые знания, умения, в том числе в области, отличной от профессиональной	УК-1	РБ СД МЦ	Знает основные способы познания, их эволюцию Умеет применять знания основных предметных областей в рамках профессиональной деятельности Имеет представление об основных методах и способах получения новых знаний	Лекции, подготовка к практическим занятиям, работа на практических занятиях, индивидуальная самостоятельная работа	Домашние задания, письменный зачет, письменный экзамен

<p>Способен вести исследовательскую деятельность, включая анализ проблем, постановку целей и задач, выделение объекта и предмета исследования, выбор способа и методов исследования, а также оценку его качества</p>	<p>УК-6</p>	<p>РБ СД МЦ</p>	<p>Знает различные методы поиска информации Умеет использовать информационно-компьютерные системы для поиска и обработки информации при мобильной разработке Владеет навыками и опытом поиска и обработки информации, необходимой при ИТ разработке</p>	<p>Лекции, подготовка к практическим занятиям, работа на практических занятиях, индивидуальная самостоятельная работа</p>	<p>Домашние задания, письменный зачет, письменный экзамен</p>
<p>Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</p>	<p>ОПК-2</p>	<p>РБ СД МЦ</p>	<p>Знание основных методологий программирования Умение формализовать прикладную задачу и выбрать соответствующую методологию программированию, наиболее подходящую для решения данной задачи Владение навыками алгоритмизации и реализации алгоритмов на языках программирования с использованием различных методологий программирования</p>	<p>Лекции, подготовка к практическим занятиям, работа на практических занятиях, индивидуальная самостоятельная работа</p>	<p>Домашние задания, письменный зачет, письменный экзамен</p>
<p>Способен применять и модифицировать математические модели для решения задач в области профессиональной деятельности</p>	<p>ОПК-3</p>	<p>РБ СД МЦ</p>	<p>Знание основных библиотек языков Kotlin и Java Умение использовать данные библиотеки при решении прикладных задач Владение навыками разработки собственных библиотек, необходимых для решения различных прикладных задач</p>	<p>Лекции, подготовка к практическим занятиям, работа на практических занятиях, индивидуальная самостоятельная работа</p>	<p>Домашние задания, письменный зачет, письменный экзамен</p>

Способен решать задачи профессиональной деятельности с использованием существующих информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-4	РБ	Знание основных принципов программирования на языке Kotlin Умение создавать приложения с использованием языка Java Владение методами отладки, редактирования и разработки программ на языке Java	Лекции, подготовка к практическим занятиям, работа на практических занятиях, индивидуальная самостоятельная работа	Домашние задания, письменный зачет, письменный экзамен
		СД			
		МЦ			

#### 4. Место дисциплины в структуре образовательной программы.

Для программы «Прикладная математика и информатика» направления подготовки 01.03.02 «Прикладная математика и информатика» настоящая дисциплина относится к циклу дисциплин майнор.

Курс рассчитан на 60 часов аудиторной нагрузки, из них 40 часов лекций, общим объемом 5 зачетных единиц (190 часов).

#### 5. Тематический план учебной дисциплины.

№	Название раздела	Всего часов	Аудиторные часы			Самостоятельная работа
			Лекции	Семинары	Практические занятия	
1	Основы программирования на Java/Kotlin	90	20	10	-	60
2	Разработка Android-приложений	100	20	10	-	70
ИТОГО		190	40	20	-	130

#### 6. Содержание дисциплины

<u>Раздел 1</u> Основы программирования на Java/Kotlin	
Тема 1	Введение в Java Примитивные типы. Особенности ООП. Исключения. Пакет Java.IO
Тема 2	Особенности языка Внутренние и вложенные классы. Дженерики и шаблоны - достоинства и недостатки. Стандартные коллекции.

<u>Раздел 2</u> Разработка Android-приложений	
Тема 1	Графический интерфейс Основные идеи UI, UIX Принципы построения интерфейса. Минимальные действия со стороны пользователя для реализации сценариев использования приложением. Обратная связь от приложения. Принципы и ограничения, диктуемые конкретным устройством. Понятие «нативности» внешнего вида. Инструменты разработки мобильного UI
Тема 2	Android Studio Особенности разработки для мобильных устройств. Ограничения мобильной разработки. Сборка приложения, запуск и отладка
Тема 3	Графические компоненты Android
Тема 4	Разработка приложения под Android Жизненный цикл приложения, взаимодействие с Google API (календарь, поиск, Google Drive), создание собственного приложения

## 7. Оценочные средства для текущего контроля и аттестации студента

### 7.1. Формы контроля знаний студентов.

Тип контроля	Форма контроля	3 курс		Параметры
		1 модуль	2 модуль	
Текущий	Домашнее задание №1	*		Письменное домашнее задание
	Курсовой проект		*	Письменное домашнее задание
Итоговый	Письменный экзамен		*	Экзамен в письменной форме

### 7.2. КРИТЕРИИ ОЦЕНКИ ЗНАНИЙ, НАВЫКОВ

Текущий контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств.

#### ДОМАШНЕЕ ЗАДАНИЕ №1

*Домашнее задание №1 выдается студентам в одном варианте. Срок выполнения домашнего задания - 2 недели. Форма представления обучающимися домашнего задания - программа, написанная на языке Java.*

#### **Пример домашнего задания №1:**

- Необходимо создать класс SmartList, реализующий интерфейс java.util.List.

```
public class SmartList<E> implements List<E> {
    ...
}
```

- Основная идея — реализовать изменяемый список так, чтобы он был оптимизирован для хранения небольшого количества элементов, но при этом в него можно было добавить произвольное их число.
- В классе должно быть объявлено ровно два поля:
  - типа `int` для хранения размера
  - ссылочного типа для хранения данных
- В зависимости от текущего размера списка в поле для данных хранятся значения разных типов:
  - Если список пуст, то значение этого поля должно быть `null`
  - Если в списке ровно один элемент, то в этом поле должна храниться ссылка на него
  - Если в списке от 2 до 5 элементов, то в этом поле хранится массив размера 5, элементы которого указывают на соответствующие элементы списка
  - Иначе в этом поле хранится ссылка на обычный `ArrayList`, в котором хранятся все элементы
- В классе должно быть определено два конструктора:
  - По умолчанию (без параметров), создает пустой список
  - Принимающий один параметр — коллекцию, элементы которой должны быть добавлены в список
- При реализации рекомендуется наследовать ваш класс от `AbstractList` (хотя он добавляет еще одно поле), и нельзя наследоваться от `ArrayList`.
- Стандартные операции коллекций (`add/contains/remove` и т.д.) должны работать за то же время, что и аналогичные методы в стандартной реализации `ArrayList`.

### Критерии оценивания и шкала оценки домашнего задания №1

Оценка	Критерии выставления оценки
«Отлично» (8-10)	Программа решает поставленную задачу, имеет корректное оформление согласно кодовым соглашениям и содержит не более 2 архитектурных, смысловых или программных ошибок.
«Хорошо» (6-7)	Программа решает поставленную задачу, имеет корректное оформление согласно кодовым соглашениям, имеет 3-4 архитектурных, смысловых или программных ошибки
«Удовлетворительно» (4-5)	Программа решает поставленную задачу, имеет корректное оформление согласно кодовым соглашениям, имеет 5 архитектурных, смысловых или программных ошибок
«Неудовлетворительно» (0-3)	Программа не решает поставленную задачу и/или не имеет корректного оформления согласно кодовым соглашениям и/или содержит более 5 архитектурных, смысловых или программных ошибок.

### КУРСОВОЙ ПРОЕКТ

*Курсовой проект выдается студентам в семи вариантах, состоит из 5 заданий. Каждому заданию присвоен свой балл. Срок выполнения курсового проекта - 5 недель. Форма представления обучающимися курсового проекта - приложение под Android.*

#### Примерный перечень тем курсового проекта:

1. «Разработка гео-ориентированного приложения»
2. «Разработка приложения с дополненной реальностью»

3. «Разработка приложения, активно использующего сенсоры»
4. «Разработка клиентского приложения для существующих Интернет-сервисов»
5. «Разработка двухмерного игрового приложения »
6. «Разработка трехмерного игрового приложения»
7. «Портирование существующего приложения»

Задачи, решаемые обучающимися при выполнении проекта (работы):

Задача 1. [2 балла] Подготовка макета и сценария использования.

Задача 2. [2 балла] Создание черновой версии UI.

Задача 3. [2 балла] Модификация UI для отображения фиксированных данных из приложения.

Задача 4. [2 балла] Модификация UI для ввода данных в приложение.

Задача 5. [2 балла] Подготовка окончательной версия приложения.

### Критерии оценивания и шкала оценки курсового проекта

Оценка	Критерии выставления оценки
«Отлично» (8-10)	Решено задач на 8 или более баллов
«Хорошо» (6-7)	Решено задач на 6-7 баллов
«Удовлетворительно» (4-5)	Решено задач на 4-5 баллов
«Неудовлетворительно» (0-3)	Решено задач на менее чем 4 балла

### ИТОГОВЫЙ КОНТРОЛЬ

Проверка качества освоения дисциплины производится в форме письменного экзамена.

### ПИСЬМЕННЫЙ ЭКЗАМЕН

*Письменный экзамен проводится в форме ответов на вопросы экзаменационного билета. Экзаменационный билет представляет собой один из алгоритмов, которые надо было реализовать в домашнем задании. На подготовку ответа выделяется 2,5 часа.*

### Критерии оценивания и шкала оценки устного экзамена

Оценка	Критерии выставления оценки
«Отлично» (8-10)	Дан развернутый ответ на поставленный вопрос. Материал изложен последовательно. Имеются логичные и аргументированные выводы.
«Хорошо» (6-7)	Дан развернутый ответ на поставленный вопрос. Материал изложен непоследовательно. Имеются логичные и аргументированные выводы.
«Удовлетворительно» (4-5)	Ответ на вопрос не является полным. Материал изложен непоследовательно. Отсутствуют выводы.

«Неудовлетворительно» (0-2)	Ответ на вопрос является неверным. Материал изложен непоследовательно. Отсутствуют выводы.
--------------------------------	--

### 7.3 Порядок формирования оценок по дисциплине

Преподаватель учитывает оценку за текущий контроль (домашние задания).

$$O_{\text{накопленная}} = 0,5 * O_{\text{д/зл}} + 0,5 * O_{\text{курсовой}}$$

Действует способ округления накопленной оценки текущего контроля в пользу студента (при значениях от 0,1 до 0,4 оценка округляется в меньшую сторону, от 0,5 до 0,9 – в большую)

Результирующая оценка за дисциплину рассчитывается следующим образом:

$$O_{\text{Результирующая}} = 0,5 O_{\text{накопленная}} + 0,5 O_{\text{экзамен}}$$

На экзамене студенту не предоставляется возможность получить дополнительный балл для компенсации оценки за текущий контроль.

## 8. Учебно-методическое и информационное обеспечение дисциплины

### 8.1 Основная литература

1. Starting with Android / М. М. Sharma and Rashmi Aggarwal. BPB Publications, 2018
2. Java for Android / By: Budi Kurniawan. Brainy Software Corp. 2014

### 8.2 Дополнительная литература

1. Соколова В.В Разработка мобильных приложений: Учебное пособие /В.В. Соколова - Томск:Изд-во Томского политех. университета, 2014. - 176 с.
2. Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебное пособие для прикладного бакалавриата / В. В. Соколова. — М. : Издательство Юрайт, 2018. — 175 с.
3. Android Design Patterns: Interaction Design Solutions for Developers / Greg Nudelman. John Wiley & Sons. 2013

## 9. Перечень учебно-методического обеспечения для самостоятельной работы, обучающихся по дисциплине.

Самостоятельная работа может рассматриваться как организационная форма обучения - система педагогических условий, обеспечивающих управление учебной деятельностью по освоению знаний и умений в области учебной деятельности без посторонней помощи. Студенту нужно четко понимать, что самостоятельная работа – не просто обязательное, а необходимое условие для получения знаний по дисциплине и развитию компетенций, необходимых в будущей профессиональной деятельности.

Самостоятельная работа проводится с целью:

- систематизации и закрепления полученных на лекциях теоретических знаний;
- углубления и расширения теоретических знаний;
- формирования умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развития познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- формирования практических (общеучебных и профессиональных) умений и навыков;

- развития исследовательских умений;
- получения навыков эффективной самостоятельной профессиональной (практической и научно-теоретической) деятельности.

В учебном процессе выделяют два вида самостоятельной работы:

- аудиторная;
- внеаудиторная.

Аудиторная самостоятельная работа по дисциплине выполняется на учебных занятиях под непосредственным руководством преподавателя и по его заданию.

Внеаудиторная самостоятельная работа - планируемая учебная работа студентов, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Самостоятельная работа, не предусмотренная программой учебной дисциплины, раскрывающей и конкретизирующей ее содержание, осуществляется студентом инициативно, с целью реализации собственных учебных и научных интересов.

Для более эффективного выполнения самостоятельной работы по дисциплине преподаватель рекомендует источники для работы, характеризует наиболее рациональную методику самостоятельной работы, демонстрирует ранее выполненные студентами работы и т. п.

Виды заданий для внеаудиторной самостоятельной работы, их содержание и характер могут иметь вариативный и дифференцированный характер, учитывать индивидуальные особенности студента.

Самостоятельная работа может осуществляться индивидуально или группами студентов online и на занятиях в зависимости от цели, объема, конкретной тематики самостоятельной работы, уровня сложности.

Контроль результатов внеаудиторной самостоятельной работы осуществляется в пределах времени, отведенного на обязательные учебные занятия по дисциплине на семинарских или контрольных занятиях.

Для представления результатов самостоятельной работы рекомендуется:

Составить план выступления, в котором отразить тему, самостоятельный характер проделанной работы, главные выводы и/или предложения, их краткое обоснование и практическое значение – с тем, чтобы в течение 3 – 5 минут представить достоинства выполненного самостоятельно задания.

Подготовить иллюстративный материал виде презентации для использования во время представления результатов самостоятельной работы в аудитории. Конкретный вариант наглядного представления результатов определяется форматом аудиторного занятия и задания преподавателя.

## **10. Материально-техническое обеспечение дисциплины и информационные технологии, используемые при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения информационных справочных систем (при необходимости).**

Для проведения всех занятий используется проектор и компьютер для проекции слайдов.