


Учреждение образования
«Белорусский государственный экономический университет»

УТВЕРЖДАЮ

Первый проректор учреждения
образования «Белорусский
государственный экономический
университет»


Е.Ф.Киреева
2022 г.
Регистрационный № УД 5798-22/уч.

ПРОМЫШЛЕННОЕ ПРОГРАММИРОВАНИЕ

Учебная программа учреждения высшего образования
по учебной дисциплине для специальности
1-31 03 06 Экономическая кибернетика (по направлениям)

Учебная программа составлена на основе ОСРБ 1-31 03 06-2013 и типового учебного плана учреждения высшего образования по специальности 1-31 03 06 «Экономическая кибернетика» (по направлениям), дата утверждения 30.06.2021, регистрационный номер № G 31-1-028/пр.-тип.

СОСТАВИТЕЛИ:

Кравчук А.С., профессор кафедры экономической информатики учреждения образования «Белорусский государственный экономический университет», доктор физ.-мат. наук, доцент

РЕЦЕНЗЕНТЫ:

Кремень Ю.А., доцент кафедры веб-технологий и компьютерного моделирования учреждения образования «Белорусский государственный университет», кандидат физ.-мат. наук, доцент;

Говядинова Н.Н., доцент кафедры информационных технологий учреждения образования «Белорусский государственный экономический университет», кандидат экономических наук, доцент.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой экономической информатики учреждения образования «Белорусский государственный экономический университет»
(протокол № 12 от 24.05.2022 г.)

Научно-методическим советом учреждения образования «Белорусский государственный экономический университет»
(протокол № 7 от 15.06.2022)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Промышленное программирование - это современный неустоявшийся термин, обозначающий совокупность методов и средств, позволяющих программистам в составе группы быстро создавать интуитивно понятные программные продукты с развитым интерфейсом пользователя, концептуально собирая стандартные объекты интерфейса, средства обработки и управления программным обеспечением таким же образом, как аппаратные объекты собираются на заводской производственной линии.

В этом смысле промышленное программирование прежде всего опирается на использование мульти-парадигменного подхода, основанного на использовании прежде всего принципов объектно-ориентированного и функционального программирования.

Таким образом для ведения занятий по данному курсу может использоваться любой из объектно-ориентированных языков, однако в связи со всеобщей современной тенденцией создания кроссплатформенных приложений представляется целесообразным использовать в данном курсе в качестве базового язык Java, получивший наибольшее распространение.

Учебная дисциплина «Промышленное программирование» относится к модулю «Программирование» и изучается в тесной связи с учебными дисциплинами «Технологии программирования», «Компьютерные сети и управление базами данных» и «Компьютерное моделирование экономических систем».

Цель учебной дисциплины – получение знаний о современном объектно-ориентированном языке программирования Java и получение практических навыков по разработке программ с его использованием.

В результате изучения учебной дисциплины выпускник должен обладать следующей компетенцией:

– БПК-2. Строить, анализировать и тестировать алгоритмы и программы решения типовых задач обработки информации с использованием структурного, объектно-ориентированного и иных парадигм программирования.

В результате изучения учебной дисциплины студент должен:

знать:

– основные понятия: алгоритмы, базовые конструкции для записи алгоритмов, простейшие типы данных;

– современный язык кроссплатформенного программирования (Java).

- способы организации ввода и вывода информации;
- простейшие алгоритмы обработки данных и особенности их реализации на языке Java;
- объектно-ориентированного программирования;

уметь:

- реализовать алгоритмы, возникающие в задачах обработки данных в области экономики;
- разрабатывать кроссплатформенное программное обеспечение для решения практических задач организаций;

владеть:

- навыками планирования разработки программ на языке Java;
- навыками написания программ на языке Java.

В соответствии с учебным планом специальности 1-31 03 06 «Экономическая кибернетика» (по направлениям) учебная программа рассчитана на 216 часов, из них аудиторных занятий 136 часов, в том числе: лекций – 68 часов, в т.ч. 24 часа УСРС, лабораторных занятий – 68 часов, в т.ч. 24 часа УСРС.

Форма получения высшего образования - очная.

Форма текущей аттестации – зачет и экзамен.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Тема 1. ОБЩИЕ СВЕДЕНИЯ О JAVA

Появление и становление Java. Виртуальная машина. Java в сравнении с другими языками. Методологии программирования, лежащие в основе Java. Инкрементная разработка. Управление памятью. Обработка ошибок. Поток. Масштабируемость. Безопасность реализации. Верификатор. Диспетчеры безопасности. Обзор online-компиляторов для Java.

Тема 2. ОСНОВЫ СИНТАКСИСА JAVA

Алфавит языка Java. Идентификаторы. Служебные слова. Лексемы. Комментарии. Типы данных. Базовые типы. Строки в Java. Переменные. Объявление переменной. Инициализация переменных. Ключевое слово var. Общие сведения о константах. Неименованные константы. Именованные константы. Перечисления enum. Простейшая программа на Java. Инструкции. Блоки. Выполнение программы. Метод main(). Общие сведения по импорту пакетов и классов. Простейшие средства ввода-вывода языка Java. Вывод на консоль. Форматированный вывод. Ввод с консоли.

Операции. Унарные и бинарные операции: арифметические, сравнения, логические операции, побитовые операции, операции присваивания и составного присваивания. Приоритет операций для переменных и констант базовых типов. Преобразование базовых типов данных: явное и неявное приведение типов, автоматические преобразования, автоматические преобразования с возможной потерей точности. Преобразование типов при операциях.

Дополнительные сведения. Большие числа BigInteger и BigDecimal: операции с объектами классов; управление округлением BigDecimal; сравнение, преобразование объектов классов BigDecimal и BigInteger. Математические вычисления и класс Math. Класс LocalDate.

Тема 3. ОПЕРАТОРЫ УПРАВЛЕНИЯ ПРОГРАММОЙ

Область видимости переменных. Условные операторы: оператор if(); оператор выбора switch(); требования к оформлению условных операторов. Операторы циклов: оператор цикла for; оператор цикла while; схема бесконечного цикла; оператор цикла do while; вложенные циклы; требования к оформлению операторов циклов. Операторы перехода.

Организация вычислений с точностью: суммирование отрезка ряда с точностью; точность вычислений согласно рекуррентным уравнениям.

Тема 4. МАССИВЫ. СОРТИРОВКИ. СТРОКИ

Одномерные массивы: длина одномерного массива; примеры простейших действий над одномерными массивами; конструкция `foreach`; интерактивный ввод значений одномерного массива; применение генератора случайных чисел для автоматического заполнения одномерного массива.

Многомерные массивы: построчный вывод на экран инициализированного двумерного массива; определение количества элементов многомерного массива по каждой из размерностей; использование цикла `foreach` для построчного вывода на экран двумерного массива; переопределение размерностей двумерного массива; примеры элементарных действий с двумерным массивом; интерактивный ввод двумерного массива; заполнение двумерного массива случайными числами.

Класс `Arrays`.

Строки: использование операции проверки равенства в случае объектов строкового типа; особенности инициализации строк с помощью литералов; простейшие методы работы со строками; методы сравнения содержания строк; интернирование строк; некоторые методы и примеры обработки строк; классы `StringBuffer` и `StringBuilder`.

Классы-обертки базовых типов. Сравнение эффективности применения базовых типов и их оберток

Тема 5. МЕТОДЫ

Введение. Заголовок, сигнатура и контракт метода. Определение (объявление) метода. Вызов метода. Тип возвращаемого значения `void`. Игнорирование в Java значений, возвращаемых методами. Возвращение в вызывающий метод результата вычислений, выполненных в вызываемом методе. Методы со списком параметров переменной длины. Использование квалификатора `final` в параметрах методов. Рекурсия. Требования к синтаксису методов, создаваемых программистом. Перегрузка методов (`overloading`).

Массивы в параметрах методов: методы и одномерные массивы; случайная генерация одномерного массива; заполнение одномерного

массива с клавиатуры; метод выбора минимума в одномерном массиве; использование константных ссылок в параметрах методов для обработки одномерных массивов; дополнительные примеры обработки массивов методами, не возвращающими результата; обработка строки с помощью массива символов; автоупаковка и автораспаковка при передаче в метод массивов примитивных типов и классов-обертки; обработка двумерных массивов в методах.

Parsing строк. Пример методов, подготавливающих число-строку к переводу в числовой формат

Стек и куча. Введение в обработку исключений

Тема 6. ОСНОВЫ ООП

Введение. Объявление (определение) класса. Объявление объекта класса и ссылки на объект класса. Инкапсуляция и сокрытие данных. Модификаторы доступа. Доступ к компонентам класса. Конструкторы класса. Деструктор класса. Ключевое слово `this`. Инициализаторы. Константные члены классов. Массивы как свойства классов. Примеры.

Объекты как параметры методов: применение классов для сокращения количества параметров в методах; передача значений по ссылке; работа с битовыми масками.

Организация списков на Java: однонаправленные (односвязные) списки; общие сведения; односвязный список объектов; двусвязные списки объектов.

Статические члены классов: статические свойства; статические константы; статические инициализаторы; статические методы.

Объекты, как свойства классов. Основы построения диаграмм классов. Использование спецификатора `final` при объявлении объекта класса.

Вложенные (внутренние) классы: собственно вложенные (внутренние) классы; особенности работы с вложенными (внутренними) классами; вложенные (внутренние) классы в локальном методе (локальные классы); статические вложенные (внутренние) классы.

Наследование: общие представления о наследовании; создание дочернего класса; права доступа при наследовании; конструкторы при наследовании; применение ключевого слова `super` для дифференциации членов класса; переопределение (`overriding`) методов; запрет наследования классов и переопределения методов; динамическая диспетчеризация методов; абстрактные классы; обобщенное понятие полиморфизма; класс `Object` и его методы; иерархия наследования и преобразование типов;

наследование в диаграммах классов; списки из объектов различных классов; ключевое слово `instanceof`.

Интерфейсы: схожесть класса и интерфейса; отличия класса от интерфейса; объявление интерфейсов; реализация интерфейса; доступ к реализациям объектов через ссылки на интерфейс; вложенные (внутренние) интерфейсы; расширение интерфейсов; множественная реализация интерфейсов в классах; методы по умолчанию; статические методы; приватные методы; константы в интерфейсах; интерфейсы как параметры и результаты методов; интерфейсы тегов (интерфейсы-маркеры); функциональные интерфейсы.

Интерфейсы в механизме обратного вызова: анонимные классы, анонимное переопределение реализации одного внешнего класса для свойства другого внешнего класса; локальное анонимное переопределение внешнего класса внутри метода другого класса; реализация интерфейса с помощью анонимного класса.

Перечисления `enum`: метод `toString()`; метод `valueOf()`; перечисления в классах; `enum` в операторе `Switch()`; конструкторы, поля и методы перечисления; ограничения на перечисления.

Записи (`record`-ы): конструктор `record`; переопределение методов; ограничения на использование `record`-ов.

Исключения: базовые сведения по исключениям в Java; общая идеология обработки исключений; формальный синтаксис исключений; простейшие примеры обработки исключений; обработка нескольких исключений; оператор `throw`; цепочка исключений; обработка исключений в методах; пробрасывание исключения в точку вызова метода; формат оператора `throws`; пример генерации в методе исключения; краткий список встроенных исключений; создание собственного класса, описывающего исключение; перехват произвольных исключений.

Ссылочные типы и клонирование объектов. Конструктор копирования. Клонирование объектов.

Иммутабельность (неизменяемость) классов и их объектов: пример неизмутабельного финального класса без сеттеров; пример построения иммутабельного класса без использования переопределенного метода `clone()`; применение метода `clone()` при построении иммутабельного класса.

Объекты не имеющие состояния (`statless object`). Пакеты.

Тема 7. ПАРАМЕТРИЗАЦИЯ

Параметризованные методы классов: методы с частично параметризованными типами; параметризация конструктора; перегрузка параметризованных методов; удобство применения параметризованных методов; недостатки параметризованных методов.

Параметризованные классы: бриллиантовая операция (Diamond operator); соглашение об именовании переменных типа; параметризованный класс с экземплярными методами; статические методы в параметризованном классе; метасимвольный аргумент (wildcards или подстановочный символ/знак); класс с несколькими параметрами; использование в качестве параметра класса параметризованного типа; сырой тип (Raw type); наследование в обобщенных типах; обобщенные типы в качестве ограничений; преимущества использования параметризованных классов.

Параметризация интерфейса: реализация параметризованного интерфейса; реализация параметризованным классом интерфейса с большим числом параметров чем у класса; доступ к реализациям объектов через ссылки на параметризованный интерфейс; вложенные (внутренние) параметризованные интерфейсы; расширение параметризованных интерфейсов; реализация множества параметризованных интерфейсов в классах; параметризованные методы «по умолчанию»; параметризованные статические методы; приватные параметризованные методы.

К вопросу о приведении типов при реализации обобщений.

Дополнительные сведения по обобщениям: выведение типов; целевые типы; стирание типа (Type Erasure).

Тема 8. КОЛЛЕКЦИИ

Общие сведения об иерархии коллекций и их интерфейсов. Абстрактные классы, частично реализующие параметризованные интерфейсы. Некоторые коллекции расширяющие абстрактные классы.

Общие сведения о шаблонах проектирования

Параметризованные интерфейсы Iterator и Iterable. Базовый интерфейс Collection. Коллекции, реализующие интерфейс List. Интерфейс ListIterator. Интерфейсы Comparable и Comparator. Коллекции, реализующие интерфейс Queue. Коллекции, реализующие интерфейс Deque. Коллекции, реализующие Deque vs коллекции, реализующие только List. Общие сведения о хэш-таблицах. Коллекции, реализующие интерфейс Map. Коллекции, реализующие интерфейс Set.

Тема 9. ЛЯМБДА-ВЫРАЖЕНИЯ

Порядок объявления и использования лямбда-выражения. Отложенное выполнение. Передача параметров в лямбда-выражение. Терминальные лямбда-выражения. Блоки кода в лямбда-выражениях. Лямбды и локальные переменные. Обобщенный (параметризованный) функциональный интерфейс. Лямбды как параметры методов. Ссылки на метод как параметры методов. Ссылки на конструкторы. Лямбды как результат методов. Встроенные функциональные интерфейсы.

Тема 10. ПОТОКИ ВВОДА-ВЫВОДА

Потоки ввода-вывода: потоки байтов; класс `InputStream`; класс `OutputStream`; абстрактные классы `Reader` и `Writer`. Чтение и запись файлов. `FileInputStream` и `FileOutputStream`. Закрывание потоков. Класс `Scanner`. Работа с `FileWriter` и `FileReader`. Буферизованные потоки. Сериализация/десериализация. Интерфейс `Serializable`. Класс `File`. Работа с ZIP-архивами.

Тема 11. STREAM API

Создание потока данных. Основы работы с ним: создание потока из коллекции; перебор элементов коллекции (метод `forEach()`); фильтрация (метод `filter()`); особенности использования промежуточных и терминальных методов. Пример фильтрации с более сложными данными. Сортировка. Промежуточные методы для получения подпотока и объединение потоков. Терминальные методы, возвращающие значения. Возвращение потока в виде коллекции. Метод `collect()`.

Требования к курсовой работе

Курсовой работы не предусмотрено.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ
«ПРОМЫШЛЕННОЕ ПРОГРАММИРОВАНИЕ»
 для дневной формы получения высшего образования по специальности
1-31 03 06 Экономическая кибернетика (по направлениям)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов						Иное*	Форма контроля знаний	
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Количество часов УСР				
						Л	Пз			Лаб
1	Общие сведения о Java					4			Тест	
2	Основы синтаксиса Java	6						2	[1-5]	Отчет, содержащий программы в соответствии с инд. заданием
3	Операторы управления программой	2						4	[1-5]	Тест + контрольный запуск заданной программы
4	Массивы. Сортировки. Строки	10						2	[1-5]	Тест + контрольный запуск заданной программы
5	Методы	4						16	[1-5]	Тест + контрольный запуск заданной программы
6	Основы ООП	18			24				[1-5]	Тест + контрольный запуск заданной программы
7	Параметризация	4			6				[1, 2]	Тест + контрольный запуск заданной программы
8	Коллекции				8	14			[1, 2]	Тест + контрольный запуск заданной программы
9	Лямбда-выражения				6	2			[1, 2]	Тест
10	Потоки ввода-вывода					2			[1, 2]	Тест + отчет, содержащий программы в соответствии с инд. заданием

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов						Иное*	Форма контроля знаний	
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Количество часов УСР				
						Л	Пз			Лаб
11	Stream API					2			[1, 2]	Тест + отчет, содержащий программы в соответствии с инд. заданием
	Всего часов	44			44	24		24		

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Методические рекомендации по организации самостоятельной работы студентов по учебной дисциплине «Промышленное программирование»

В овладении знаниями учебной дисциплины важным этапом является самостоятельная работа студентов. Рекомендуется бюджет времени для самостоятельной работы в среднем 1,5-2 часа на 2-х часовое аудиторное занятие.

Основными направлениями самостоятельной работы студента являются:

- первоначально подробное ознакомление с программой учебной дисциплины;

- ознакомление со списком рекомендуемой литературы по учебной дисциплине в целом и ее разделам, наличие ее в библиотеке и других доступных источниках, изучение необходимой литературы по теме, подбор дополнительной литературы;

- изучение и расширение лекционного материала преподавателя за счет специальной литературы, консультаций;

- подготовка к лабораторным занятиям по специально разработанным планам с изучением основной и дополнительной литературы;

- подготовка к выполнению диагностических форм контроля (контрольные задания на компьютере);

- подготовка к экзамену.

Примерный перечень тем управляемой самостоятельной работы обучающихся:

- общие сведения о Java;
- основы синтаксиса Java;
- операторы управления программой;
- массивы, сортировки, строки;
- методы;
- коллекции;
- лямбда-выражения;
- потоки ввода-вывода;
- Stream API.

Средства диагностики результатов учебной деятельности:

- краткий опрос по синтаксису в соответствии с материалами прошедших лекций и лабораторных работ перед началом занятий в подгруппах;

ЛИТЕРАТУРА

Основная:

1. Шилдт, Г. Java. Полное руководство : [полностью обновлено с учетом версии Java SE 9 (JDK 9) : исчерпывающее описание языка программирования Java : пер. с англ.] / Г. Шилдт. - 10-е изд. - Москва ; Санкт-Петербург : Диалектика, 2018. - 491 с. : ил.


2. Гниденко, И. Г. Технологии и методы программирования : учебное пособие для студентов высших учебных заведений, обучающихся по инженерно-техническим направлениям / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров ; [Санкт-Петербургский гос. экон. ун-т]. - Москва : Юрайт, 2021. - 233, [2] с. : ил.

Дополнительная литература:

3. Болбот, О. М. Классы в языке программирования Java [Электронный ресурс]: учебно-методическое пособие для студентов и слушателей системы повышения квалификации и переподготовки / О. М. Болбот, В. В. Сидорик ; под общ. ред. В. В. Сидорика ; Белорусский национальный технический университет. – Минск : БНТУ, 2020. – 76 с. – Режим доступа: <https://rep.bntu.by/handle/data/70419>. – Дата доступа: 02.06.2022.

4. Гуськова, О. И. Объектно ориентированное программирование в Java : учебное пособие / О. И. Гуськова. – Москва : Московский педагогический государственный университет (МПГУ), 2018. – 240 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=500355> (дата обращения: 02.06.2022). – Библиогр. в кн. – ISBN 978-5-4263-0648-6. – Текст : электронный.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УВО

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Эконометрика	Математических методов в экономике	Предложений нет  Г.О. Читая	Протокол № 12 от «24» мая 2022 г.

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ УВО
на ____ / ____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры экономической информатики (протокол № ____ от _____ 201_ г.)

Заведующий кафедрой
к.э.н., доцент

_____ А.М. Зеневич
(подпись) (И.О.Фамилия)

УТВЕРЖДАЮ
Декан факультета ЦЭ
к.э.н., доцент

_____ Т.Н. Налецкая
(подпись) (И.О.Фамилия)