

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

Учебно-методическое объединение по образованию  
в области информатики и радиоэлектроники

**УТВЕРЖДАЮ**

Первый заместитель Министра образования  
Республики Беларусь

\_\_\_\_\_ А.Г.Баханович

\_\_\_\_\_

Регистрационный № ТД-\_\_\_\_\_/тип.

**ИЗБРАННЫЕ ГЛАВЫ ИНФОРМАТИКИ**

Типовая учебная программа по учебной дисциплине  
для специальности

**1-40 04 01 Информатика и технологии программирования**

**СОГЛАСОВАНО**

Председатель Учебно-методического  
объединения по образованию в  
области информатики и  
радиоэлектроники

\_\_\_\_\_ В.А. Богуш

\_\_\_\_\_

**СОГЛАСОВАНО**

Начальник Главного управления  
профессионального образования  
Министерства образования  
Республики Беларусь

\_\_\_\_\_

\_\_\_\_\_

**СОГЛАСОВАНО**

Проректор по научно-методической  
работе Государственного учреждения  
образования «Республиканский  
институт высшей школы»

\_\_\_\_\_ И.В. Титович

\_\_\_\_\_

Эксперт-нормоконтролер

\_\_\_\_\_

\_\_\_\_\_

Минск 2023

**СОСТАВИТЕЛИ:**

А.В.Жвакина, доцент кафедры информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент

**РЕЦЕНЗЕНТЫ:**

Кафедра информационных технологий и математики учреждения образования «БИП – университет права и социально-информационных технологий» (протокол № 6 от 16.01.2023);

В.А.Полубок, ведущий инженер-программист Республиканского унитарного предприятия «Центр информационных технологий Национального статистического комитета Республики Беларусь», кандидат технических наук, доцент

**РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ В КАЧЕСТВЕ ТИПОВОЙ:**

Кафедрой информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 4 от 14.12.2022);

Научно-методическим советом учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол №\_\_ от \_\_\_\_\_);

Научно-методическим советом по разработке программного обеспечения и информационно-коммуникационным технологиям Учебно-методического объединения по образованию в области информатики и радиоэлектроники (протокол № \_\_\_ от \_\_\_\_\_)

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

### ХАРАКТЕРИСТИКА УЧЕБНОЙ ДИСЦИПЛИНЫ

Типовая учебная программа по учебной дисциплине «Избранные главы информатики» разработана для студентов учреждений высшего образования, обучающихся по специальности 1-40 04 01 Информатика и технологии программирования в соответствии с требованиями образовательного стандарта высшего образования I степени и типового учебного плана вышеуказанной специальности.

Актуальность изучения учебной дисциплины заключается в необходимости подготовки квалифицированных специалистов, владеющих фундаментальными знаниями и практическими навыками в области современных средств и технологий разработки, доставки и запуска программных продуктов.

Учебная дисциплина «Избранные главы информатики» относится к модулю «Прикладная математическая подготовка» государственного компонента учебного плана.

### ЦЕЛЬ, ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Цель учебной дисциплины: ознакомление с современными средствами создания программных продуктов и технологиями разработки веб-приложений.

Задачи учебной дисциплины:

приобретение знаний в области синтаксиса языка Python, объектно-ориентированного программирования, методов работы с коллекциями, потоками команд и данных, основных шаблонов для построения программ, методики разработки с применением сред быстрой разработки приложений;

изучение принципов параллельного, сетевого и веб- программирования, доставки новых модулей разрабатываемого проекта;

приобретение навыков программирования на языке Python, использования распределенной системы контроля версий;

овладение методами контейнеризации для разработки, доставки и запуска приложений.

Базовыми учебными дисциплинами для учебной дисциплины «Избранные главы информатики» являются: «Основы алгоритмизации и программирования», «Программирование», «Конструирование программ». В свою очередь учебная дисциплина «Избранные главы информатики» является базой для такой учебной дисциплины государственного компонента, как «Модели данных и системы управления базами данных» и учебных дисциплин компонента учреждения высшего образования: «Обработка экспериментальных данных», «Интеллектуальный анализ данных», «Современные технологии разработки web-приложений».

## ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ СОДЕРЖАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

В результате изучения учебной дисциплины «Избранные главы информатики» формируются следующие компетенции:

*универсальная:* обладать навыками творческого аналитического мышления;

*базовая профессиональная:* применять математические методы и алгоритмы теоретической информатики для решения практических задач.

В результате изучения учебной дисциплины студент должен:

*знать:*

синтаксис языка Python;

методы работы с коллекциями, потоками команд и данных;

принципы объектно-ориентированного программирования, метапрограммирования;

основные шаблоны для построения приложений и веб-приложений;

основы CI/CD;

основы контейнеризации;

методику разработки с применением сред быстрой разработки приложений;

*уметь:*

строить архитектуру малых, средних и корпоративных приложений с использованием языка Python;

разрабатывать приложения на фреймворках Django/Flask;

*владеть:*

технологией контейнеризации для разработки, доставки и запуска приложений,

навыками параллельного, сетевого и веб- программирования на языке Python, применения распределенной системы контроля версий

В рамках образовательного процесса по учебной дисциплине «Избранные главы информатики» студент должен приобрести не только теоретические и практические знания, умения и навыки по специальности, но и развить свой ценностно-личностный, духовный потенциал, сформировать качества патриота и гражданина, готового к активному участию в экономической, производственной, социально-культурной и общественной жизни страны.

Типовая учебная программа рассчитана на 108 учебных часов, из них – 50 аудиторных. Примерное распределение аудиторных часов по видам занятий: лекции – 26 часов, лабораторные занятия – 24 часа.

## ПРИМЕРНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН

Наименование раздела, темы	Всего аудиторных часов	Лекции	Лабораторные занятия
Тема 1. Основы CI/CD, Git	4	2	2
Тема 2. Docker	4	2	2
Тема 3. Общая характеристика языка Python	4	2	2
Тема 4. Базовые типы данных и операции	6	4	2
Тема 5. Управляющие конструкции, функции	4	2	2
Тема 6. Итераторы, генераторы, декораторы, обработка исключений	4	2	2
Тема 7. Объектно-ориентированное программирование и классы	6	4	2
Тема 8. Файлы и сериализация данных	4	2	2
Тема 9. Модули, пакеты, стандартная библиотека. Визуализация данных	4	2	2
Тема 10. Сетевое и веб-программирование	6	2	4
Тема 11. Параллельное программирование	4	2	2
<b>Итого:</b>	<b>50</b>	<b>26</b>	<b>24</b>

## СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

### Тема 1. ОСНОВЫ CI/CD, GIT

Основы CI/CD. Платформы для настройки CI/CD. Git и системы контроля версий.

### Тема 2. DOCKER

Назначение. Основные команды. Dockerfile. Docker Compose. Docker Networking.

### Тема 3. ОБЩАЯ ХАРАКТЕРИСТИКА ЯЗЫКА PYTHON

История Python. Установка, реализации, справка. IDE для Python. Структура программы. Справка. Синтаксис Python. Линтеры. Переменные. Управление памятью.

### Тема 4. БАЗОВЫЕ ТИПЫ ДАННЫХ И ОПЕРАЦИИ

Классификация типов данных. Логический тип. NoneType. Числа. Коллекции: последовательность, списки, строки, кортежи, отображения, множества, числовой диапазон, bytearray, bytes. Функции и модули. Общие функции объектов. Поверхностное и глубокое копирование. Взаимное преобразование. Приоритет операций.

### Тема 5. УПРАВЛЯЮЩИЕ КОНСТРУКЦИИ, ФУНКЦИИ

Условный оператор. Циклы: while, for. Перемещение по коллекциям. Коллекционные включения. Прерывание и продолжение циклов. Функции. Вложенные функции. Замыкания. Области видимости. Параметры функции: позиционные и ключевые. Возврат значений. Функция как аргумент функции. Анонимные лямбда-функции. Функции как метод. Документирование кода.

### Тема 6. ИТЕРАТОРЫ, ГЕНЕРАТОРЫ, ДЕКОРАТОРЫ, ОБРАБОТКА ИСКЛЮЧЕНИЙ

Назначение, синтаксис и функции итераторов. Назначение, синтаксис и функции генераторов, генераторные выражения. Назначение, синтаксис и функции декораторов, декораторы функций, декораторы классов, параметризованные декораторы. Категории ошибок и их обработка. Понятие исключения. Конструкция try. Возбуждение исключений raise. Утверждения assert. Определение собственного типа исключения.

### Тема 7. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ И КЛАССЫ

Основные понятия и принципы объектно-ориентированного программирования (ООП). Класс. Специальные методы. Атрибуты. Наследование и полиморфизм. Множественное наследование. Метаклассы. Примеси.

## Тема 8. ФАЙЛЫ И СЕРИАЛИЗАЦИЯ ДАННЫХ

Файловый объект. Основные свойства и методы файлов. Сериализация и десериализация. CSV- и JSON-файлы.

## Тема 9. МОДУЛИ, ПАКЕТЫ, СТАНДАРТНАЯ БИБЛИОТЕКА. ВИЗУАЛИЗАЦИЯ ДАННЫХ

Модули. Пакеты. Классификация. Подключение и использование модулей и пакетов. Специальные атрибуты. Стандартная библиотека Python. Регулярные выражения. Визуализация данных.

## Тема 10. СЕТЕВОЕ И ВЕБ-ПРОГРАММИРОВАНИЕ

Основные понятия. Основные возможности для работы с сетевым стеком. HTTP, HTTPS, DNS, Django/Flask.

## Тема 11. ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

Основные понятия. Выполнение параллельных задач. Планировщик заданий. Многопоточное программирование. Очередь.

**ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ****ЛИТЕРАТУРА****ОСНОВНАЯ**

1. Прохоренок, Н. А. Python 3. Самое необходимое / Н. А. Прохоренок, В. А. Дронов. – Санкт-Петербург : БХВ-Петербург, 2016. – 464 с.
2. Седер, Н. Python : экспресс-курс / Н. Седер ; пер. с англ. Е. Матвеева. – 3-е изд. – Санкт-Петербург : Питер, 2021. – 480 с.
3. Python и DevOps : ключ к автоматизации Linux / Н. Гифт [и др.]. – Санкт-Петербург : Питер, 2023. – 544 с.
4. Доусон, М. Программируем на Python / М. Доусон. – [3-е изд.]. – Санкт-Петербург : Питер, 2022. – 416 с.
5. Плас, Дж. В. Python для сложных задач : наука о данных и машинное обучение / Дж. В. Плас. – Санкт-Петербург : Питер, 2022. – 576 с.
6. Бенгфорт, Б. Прикладной анализ текстовых данных на Python : машинное обучение и создание приложений обработки естественного языка / Б. Бенгфорт, Р. Билбро, Т. Охеда. – Санкт-Петербург : Питер, 2020. – 368 с.
7. Хайнеман, Д. Алгоритмы. Справочник с примерами на C, C++, Java и Python / Д. Хайнеман, Г. Поллис, С. Селков ; пер. с англ. – 2-е изд. – Москва ; Санкт-Петербург : Диалектика, 2020. – 432 с.
8. Бэрри, П. Изучаем программирование на Python / П. Бэрри. – Москва : Эксмо, 2018. – 624 с.
9. Мэтиз Э. Изучаем Python : программирование игр, визуализация данных, веб-приложения / Э. Мэтиз. – 2-е изд. – Санкт-Петербург : Питер, 2018. – 611 с.
10. Лутц, М. Изучаем Python / М. Лутц. – 4-е изд. – Санкт-Петербург : Символ-Плюс, 2015. – 381 с.
11. Форсье, Д. Django. Разработка веб-приложений на Python / Д. Форсье, П. Биссекс, У. Чан. – Санкт-Петербург : Символ, 2013. – 456 с.

**ДОПОЛНИТЕЛЬНАЯ**

12. Vincent, W. S. Django for Beginners / W. S. Vincent. – William S. Vincent, – 2018. – 317 p.
13. Vincent, W. S. Django for Professionals : Production websites with Python & Django / W. S. Vincent. – William S. Vincent, – 2020. – 290 p.
14. Vincent, W. S. Django for APIs: Build web APIs with Python and Django / W. S. Vincent. – William S. Vincent, – 2020. – 155 p.
15. Django documentation [Электронный ресурс] / 2005-2021 Django Software Foundation and individual contributors. – Режим доступа: <https://docs.djangoproject.com/en/3.2/>. – Дата доступа: 18.01.2023.
16. Django на русском [Электронный ресурс] / 2008-2021 Ruslan Popov Dmytro Kostochko. – Режим доступа: <https://djbook.ru/>. – Дата доступа: 18.01.2023.
17. William Vincent [Электронный ресурс] / William Vincent. – Режим доступа: <https://wsvincent.com/>. – Дата доступа: 18.01.2023.

18. Flask Documentation (1.1.x) [Электронный ресурс] / 2010 Pallets. – Режим доступа: <https://flask.palletsprojects.com/en/1.1.x/>. – Дата доступа: 18.01.2023.

19. The Flask Mega-Tutorial Part I: Hello, World! [Электронный ресурс] / 2012-2021 by Miguel Grinberg. – Режим доступа: <https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world>. – Дата доступа: 18.01.2023.

20. W3Schools Online Web Tutorials [Электронный ресурс] / 1999-2023 by Refsnes Data. – Режим доступа: <https://www.w3schools.com/>. – Дата доступа: 18.01.2023.

21. Бесплатное руководство по теме «Git» – Git: Become an Expert in Git & GitHub in 4 hours [Электронный ресурс] / Udemy. – 2021 Udemy, Inc. – Режим доступа: <https://www.udemy.com/course/git-expert-4-hours/>. – Дата доступа: 18.01.2023.

22. Git Documentation [Электронный ресурс] / – Software Freedom Conservancy. – Режим доступа : <https://git-scm.com/doc>. – Дата доступа: 18.01.2023.

## МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЩАЮЩИХСЯ

При изучении учебной дисциплины рекомендуется использовать следующие формы самостоятельной работы:

- работа с учебной и справочной литературой;
- подготовка к лабораторным работам;
- повторение пройденного теоретического материала;
- изучение статей и дополнительной литературы по отдельным темам учебной дисциплины.

## ПЕРЕЧЕНЬ РЕКОМЕНДУЕМЫХ СРЕДСТВ ДИАГНОСТИКИ КОМПЕТЕНЦИЙ СТУДЕНТА

Типовым учебным планом по специальности 1-40 04 01 Информатика и технологии программирования в качестве формы промежуточной аттестации по учебной дисциплине «Избранные главы информатики» рекомендуется зачет. Оценка учебных достижений студента производится по системе «зачтено/не зачтено».

Для текущего контроля по учебной дисциплине и диагностики компетенций студентов могут использоваться следующие формы:

- собеседования;
- контрольные опросы;
- тестирование;
- отчеты по аудиторным лабораторным работам с их устной защитой;
- оценивание на основе модульно-рейтинговой системы.

## РЕКОМЕНДУЕМЫЕ МЕТОДЫ (ТЕХНОЛОГИИ) ОБУЧЕНИЯ

Основные рекомендуемые методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

элементы проблемного обучения (проблемное изложение, частично-поисковый метод), реализуемые на лекционных и лабораторных занятиях;

элементы контролируемого обучения (контрольные опросы, контрольные работы), реализуемые на лабораторных занятиях, а также в ходе самостоятельной работы студентов.

## ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

1. Git. Docker.
2. Итераторы, генераторы, декораторы.
3. Сериализатор на Python.
4. Django/Flask.
5. Многопоточность и CI/CD.

## ПРИМЕРНЫЙ ПЕРЕЧЕНЬ КОМПЬЮТЕРНЫХ ПРОГРАММ

*(необходимого оборудования, наглядных пособий и др.)*

1. Git – <https://git-scm.com/downloads>.
2. Docker – <https://www.docker.com>.
3. Python (IDLE) – <https://python.org/downloads>.
4. Visual Studio Code.
5. Visual Studio.