

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ:
Директор



А. В. Замятин

« 14 » июня 20 23 г.

Рабочая программа дисциплины

Разработка программного обеспечения и скриптовые языки

по направлению подготовки

09.04.03 Прикладная информатика

Направленность (профиль) подготовки :

Цифровизация государственного и муниципального управления

Форма обучения

Очная

Квалификация

Магистр

Год приема

2023

Код дисциплины в учебном плане: Б1.О.03.02

СОГЛАСОВАНО:

Руководитель ОП

Н.Л.Еремина

Председатель УМК

С.П. Сущенко

Томск – 2023

1. Цель и планируемые результаты освоения дисциплины

Целью освоения дисциплины является формирование следующих компетенций:

- УК-2. - Способен управлять проектом на всех этапах его жизненного цикла;
- ОПК-2. -Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач;
- ИУК-2.3 -Обеспечивает выполнение проекта в соответствии с установленными целями, сроками и затратами..
- ИУК-2.2 -Разрабатывает программу действий по решению задач проекта с учетом имеющихся ресурсов и ограничений..
- ИУК-2.1 -Формулирует цель проекта, обосновывает его значимость и реализуемость..
- ИОПК-2.3 -Использует методы современных интеллектуальных технологий для решения профессиональных задач.
- ИОПК-2.2 -Знает современные подходы, методы применения современных интеллектуальных технологий для решения профессиональных задач.
- ИОПК-2.1-Владеет необходимыми методами алгоритмизации и программирования для решения профессиональных задач.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

- ОР-1.1.1. Уметь формулировать цель проекта, обосновывает его значимость и реализуемость.
- ОР-1.2.1. Уметь разрабатывать программу действий по решению задач проекта с учетом имеющихся ресурсов и ограничений.

ОР-1.3.1 Уметь обеспечивать выполнение проекта в соответствии с установленными целями, сроками и затратами.

ОР-2.1.1. Знать основные концепции современных вычислительных систем и программного обеспечения (в том числе отечественного производства).

ОР-2.2.1. Владеть методами высокопроизводительных вычислительных технологий, современного программного обеспечения (в том числе отечественного производства) для решения задач профессиональной деятельности.

ОР-2.3.1. Владеть инструментальными средствами высокопроизводительных вычислений в научной и практической деятельности.

ОР-4.1.1. Умеет применять знания о современных платформах и технологиях разработки в практической деятельности.

ОР-4.3.1. Умеет применять современные платформы и технологии для разработки программных систем.

ОР-6.1.1. Знает инструментальные среды, программно-технические платформы для решения профессиональных задач

ОР-6.2.1. Знает принципы разработки оригинальных программных средств для решения профессиональных задач

2. Задачи освоения дисциплины

Формирование навыков для профессионального программирования с использованием современных языков и их реализаций, встроенных и сторонних библиотек и пакетов программ; ознакомление с современными профессиональными стандартами информационных технологий, получение детальных знаний парадигм и методологий программирования, особенностей языков программирования общего и специального назначения, таких как Python, Javascript, Perl; работа с разными инструментальными системами разработки программ.

3. Место дисциплины в структуре образовательной программы

Дисциплина «Разработка программного обеспечения и скриптовые языки» относится к обязательной части «Дисциплины», входит в модуль «Разработка программного обеспечения».

4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине

Второй семестр, экзамен

5. Входные требования для освоения дисциплины

Для освоения дисциплины необходимо:

- иметь глубокие представления о современных информационных технологиях и компьютерных науках.

- обладать навыками программирования, пониманием основ алгоритмизации и знанием базовых методов работы интерпретаторов и компиляторов, знанием теории автоматов и формальных языков

Пререквизиты дисциплины: нет.

Постреквизиты дисциплины: Devops инженерия.

6. Язык реализации

Русский

7. Объем дисциплины

Общая трудоемкость дисциплины составляет 4 з.е., 144 часов, из которых:

-лекции: 16 ч.

-лабораторные: 16 ч.

в том числе практическая подготовка: 16ч.

Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины, структурированное по темам

Тема 1. Общее введение в теорию компиляции

Общее введение в теорию компиляции. Задачи и подходы трансляции.

Ликбез по компиляторам и принципам трансляции. Фазы компиляции

Тема 2. Современные языки программирования

Парадигмы и классификация языков программирования

Обзор современных языков программирования

Тема 3. Разработка ПО с использованием скриптовых языков

Основы методологии разработки ПО и организации командной работы

Основы разработки ПО на ЯП Python

Практическое использование языка Python

Основы разработки ПО на ЯП Perl

Практическое использование языка Perl

Основы разработки ПО на ЯП Javascript

Практическое использование языка Javascript

9. Текущий контроль по дисциплине

Самостоятельная работа студентов по дисциплине организуется в следующих формах:

1) изучение теоретического материала на основе рекомендуемых списков основной и дополнительной литературы, а также баз данных и информационно-справочных систем;

2) выполнение группового проекта: студенты объединяются в команды (5-7 человек) для выполнения группового проекта, каждый студент получает определенную роль (роли). В ходе выполнения проекта студент должен выполнить работы, соответствующие своей роли (ролям) и текущей фазе проекта.

Текущий контроль по лабораторным работам осуществляется в виде проверки выполнения заданий лабораторной работы. Текущий контроль успеваемости по теоретическому материалу осуществляется в виде контрольных работ.

Итоговая оценка по предмету (зачет с оценкой) выставляется следующим образом:

«отлично» – студент выполнил не менее 75% запланированных работ по групповому проекту, выполнил все лабораторные работы, нет неудовлетворительных оценок за контрольные работы, средняя (округленная) оценка за контрольные работы – «отлично»;

«хорошо» – студент выполнил не менее 75% запланированных работ по групповому проекту, выполнил все лабораторные работы, нет неудовлетворительных оценок за контрольные работы, средняя (округленная) оценка за контрольные работы – «хорошо»;

«удовлетворительно» – студент выполнил не менее 75% запланированных работ по групповому проекту, выполнил все лабораторные работы, нет неудовлетворительных оценок за контрольные работы, средняя (округленная) оценка за контрольные работы – «удовлетворительно»;

«неудовлетворительно» – студент не сдал лабораторные работы, не выполнил 75% запланированных работ по групповому проекту или сдал хотя бы одну контрольную работу на «неудовлетворительно».

Во время зачета студент может повысить свою оценку, сдав заново соответствующую контрольную работу, при условии выполнения остальных требований к оценке. Практическая подготовка оценивается по результатам выполненных лабораторных работ.

10. Порядок проведения и критерии оценивания промежуточной аттестации

Теоретические и практические результаты формируются компетенциями ИУК-2.1; ИУК-2.2; ИУК-2.3; ИОПК-2.1; ИОПК-2.2; ИОПК-2.3 и результатами обучения:

№	Этапы формирования компетенций (разделы дисциплины)	Код и наименование результатов обучения	Вид оценочного средства (тесты, задания, кейсы, вопросы и др.)
1.	Тема 1. Общее введение в теорию компиляции	ИУК-2.1; ИУК-2.2; ИУК-2.3; ИОПК-2.1; ИОПК-2.2; ИОПК-2.3	Лабораторные работы №1 - №3 Вопросы для текущего контроля
2.	Тема 2. Современные языки программирования	ИУК-2.1; ИУК-2.2; ИУК-2.3; ИОПК-2.1; ИОПК-2.2; ИОПК-2.3	Лабораторные работы №4 - №6 Вопросы для текущего контроля
3	Тема 3. Разработка ПО с использованием скриптовых языков	ИУК-2.1; ИУК-2.2; ИУК-2.3; ИОПК-2.1; ИОПК-2.2; ИОПК-2.3	Лабораторные работы №7 - №8 Вопросы для текущего контроля

Студенты объединяются в команды для выполнения группового проекта. Проект заключается в сопровождении разработки программного обеспечения для заданной предметной области (область определяется и описывается преподавателем). В ходе выполнения проекта студент выполняет работы, соответствующие текущей фазе проекта.

Выполнение проекта ведется во время лабораторных работ и во время самостоятельной работы студента.

Текущий контроль осуществляется путем проверки для каждого студента выполнения необходимых действий для текущей фазы проекта. Также проводятся контрольные работы по теоретическому материалу по части вопросов из п. 3.2 в соответствии с пройденным материалом.

11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle».

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

Самостоятельная работа студентов по предмету организуется в следующих формах:

- 1) самостоятельное изучение основного теоретического материала, ознакомление с дополнительной литературой, Интернет-ресурсами, изучение материала лекций, указаний, выданных преподавателем при проведении лабораторных работ.
- 2) выполнение практических задач в рамках предметной области с использованием изученных на лекциях технологий и программных решений.

В качестве учебно-методического обеспечения самостоятельной работы используется основная и дополнительная литература по предмету, Интернет-ресурсы,

Тема 1. Общее введение в теорию компиляции, Тема 4. Ликбез по компиляторам и принципам трансляции	Основная литература: [1] Дополнительная литература: [1] Интернет-ресурсы: [1]
---	---

Тема 2. Современные языки программирования. Тема 3. парадигмы ЯП и классификация	Основная литература: [1] Дополнительная литература: [1], [2] Интернет-ресурсы: [1], [2]
Темы 5-7. Практическое использование языков Python, Javascript, Perl	Основная литература: [2] Дополнительная литература: [2], [3] Интернет-ресурсы: [1], [2]
Тема 8. Основы методологии разработки ПО и организации командной работы	Основная литература: [3] Интернет-ресурсы: [3]

12. Перечень учебной литературы и ресурсов сети Интернет

Основная литература

1. Мозговой М.В. Классика программирования. Алгоритмы, языки, автоматы, компиляторы. Практический подход. СПб.: Наука и Техника 2006. 320с.
2. П. Н. Вабищевич. Численные методы: вычислительный практикум: [практическое применение численных методов при использовании алгоритмического языка Python] М: Ленанд 2015. – 319 с.: рис.

Дополнительная литература

3. Fernández M. Programming Languages and Operational Semantics.[Электронный ресурс] London:Springer 2014. 209 p. 10 illus.: online resource. – URL: <http://dx.doi.org/10.1007/978-1-4471-6368-8> (дата обращения: 01.11.2021).

в) ресурсы сети Интернет:

- открытые онлайн-курсы
- Журнал «Эксперт» - <http://www.expert.ru>
- Официальный сайт Федеральной службы государственной статистики РФ - www.gsk.ru
- Официальный сайт Всемирного банка - www.worldbank.org
- Общероссийская Сеть КонсультантПлюс Справочная правовая система. <http://www.consultant.ru>
- ...

13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

При осуществлении образовательного процесса используется онлайн интерпретатор языков Python3 и Javascript – Repl.it, онлайн интерпретатор языка Javascript – Tryit. Для языка Perl используется открытая и бесплатно распространяемая реализация strawberry perl 5.24.

б) информационные справочные системы:

- Электронный каталог Научной библиотеки ТГУ – <http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system>
- Электронная библиотека (репозиторий) ТГУ – <http://vital.lib.tsu.ru/vital/access/manager/Index>
- ЭБС Лань – <http://e.lanbook.com/>
- ЭБС Консультант студента – <http://www.studentlibrary.ru/>
- Образовательная платформа Юрайт – <https://urait.ru/>
- ЭБС ZNANIUM.com – <https://znanium.com/>
- ЭБС IPRbooks – <http://www.iprbookshop.ru/>

в) профессиональные базы данных (при наличии):

1. Programming Languages - commonly used features in a side-by-side format [Электронный ресурс] / Hyperpolyglot.– 2016. – URL: <http://hyperpolyglot.org/> (дата обращения: 06.10.2016).
2. The Computer Language Benchmarks Game [Электронный ресурс] / Alioth Debian .– 2016. – URL: <http://benchmarksgame.alioth.debian.org/> (дата обращения: 14.10.2016).
3. The Scrum Guide [Электронный ресурс] / Ken Schwaber .– 2017. – URL: <https://www.scrum.org/resources/scrum-guide>
4. Электронная библиотека (репозиторий) ТГУ [Электронный ресурс]. – Электрон. дан. – Томск, 2016- . URL: <http://vital.lib.tsu.ru/vital/access/manager/Index– ...>

14. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

Аудитории для проведения занятий лекционного и семинарского типа индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации в смешенном формате («Актру»).

15. Информация о разработчиках

Овсянников Михаил Сергеевич – старший преподаватель кафедры теоретических основ информатики ТГУ