

МИНОБРНАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ
Директор института прикладной
математики и компьютерных наук
А.В. Замятин
« 02 » _____ 2021 г.



Фонд оценочных средств по дисциплине

Языки программирования

Направление подготовки

10.05.01 Компьютерная безопасность

код и наименование направления подготовки

Анализ безопасности компьютерных систем

наименование специальности

Томск–2021

ФОС составил(и):
канд. физ.-мат. наук, доцент
доцент кафедры компьютерной безопасности



С.И. Самохина

Рецензент:
канд. техн. наук, доцент,
зав. кафедрой компьютерной безопасности



С.А. Останин

Фонд оценочных средств одобрен на заседании учебно-методической комиссии
института прикладной математики и компьютерных наук (УМК ИПМКН)

Протокол от 17 июня 2021 г. № 05

Председатель УМК ИПМКН,
д-р техн. наук, профессор



С.П. Сущенко

Фонд оценочных средств (ФОС) является элементом системы оценивания сформированности компетенций у обучающихся в целом или на определенном этапе ее формирования.

ФОС разрабатывается в соответствии с рабочей программой (РП) дисциплины и включает в себя набор оценочных материалов для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине.

1. Компетенции и результаты обучения, формируемые в результате освоения дисциплины

Компетенция	Индикатор компетенции	Код и наименование результатов обучения (планируемые результаты обучения, характеризующие этапы формирования компетенций)	Критерии оценивания результатов обучения			
			Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
ОПК-7. Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ИОПК-7.1 Осуществляет построение алгоритма, проведение его анализа и реализации в современных программных комплексах ИОПК-7.2 Понимает общие принципы построения и использования языков программирования высокого уровня и низкого уровня ИОПК-7.3 Демонстрирует навыки создания программ с применением методов и инструментальных средств программирования для решения различных профессиональных, исследовательских и прикладных задач ИОПК-7.4 Осуществляет обоснованный выбор инструментария программирования и	Знать: ОР-7.2.1. Освоить данные C# и Python. ОР-7.2.2. Ознакомиться с библиотеками numpy, pandas, matplotlib для работы с искусственным интеллектом на языке Python. Уметь: ОР-7.3.1. Освоить работу с online компиляторами как средствами редактирования, отладки, компиляции и выполнения программ. ОР-7.4.1. Научиться использовать библиотеки для работы с большими данными и искусственным интеллектом. ОР-7.1.1. Применять на практике структуры данных для хранения и обработки данных. ОР-7.2.3. Научиться создавать пользовательские классы на C#.	Знает основы языков программирования C# и Python. Способен составить и отладить программу. Умеет пользоваться библиотеками Python. Умеет создавать пользовательские классы на языке C#.	Знает основы языков программирования C# и Python. Способен составить и отладить программу. Умеет пользоваться библиотеками Python. Умеет создавать пользовательские классы на языке C#. Однако, при создании программы допускает неточности.	Знает основы языков программирования C# и Python. Способен составить и отладить программу. Умеет пользоваться библиотеками Python. Умеет создавать пользовательские классы на языке C#. Однако, при выборе средств и создании программ допускает значительные ошибки.	Не знает основы языков программирования C# и Python. Способен составить и отладить программу. Не умеет пользоваться библиотеками Python. Не умеет создавать пользовательские классы на языке C#.

	способов организации программ	Владеть: ОР-7.1.2. Освоить методы обработки данных различных типов. ОР-7.4.2. Изучить методы отладки и тестирования программ.				
--	-------------------------------	--	--	--	--	--

2. Этапы формирования компетенций и виды оценочных средств

№	Этапы формирования компетенций (разделы дисциплины)	Код и наименование результатов обучения	Вид оценочного средства (тесты, задания, кейсы, вопросы и др.)
1.	Введение в язык С#. Назначение и особенности данного алгоритмического языка и использование в современном информационном обществе.	ОР-7.2.1. ОР-7.3.1. ОР-7.4.2.	Задание 1, 2, 3, тест
2.	Создание классов. Конструкторы. Перегрузка операторов и методов класса.	ОР-7.1.1. ОР-7.2.3. ОР-7.1.2.	Задание 1, 2, тест
3.	Наследование.	ОР-7.1.1. ОР-7.2.3.	Задание 3, тест
4.	Коллекции и события языка С# и их применение для решения поставленных задач.	ОР-7.1.2.	Задание 3, тест
5.	Введение в язык Python. Назначение и особенности данного алгоритмического языка и использование в современном информационном обществе.	ОР-7.2.1. ОР-7.3.1. ОР-7.4.2.	Задание 4, 5. Кейсы 1, 2.
6.	Библиотеки numpy, pandas, matplotlib и другие. Использование библиотек для моделирования задач искусственного интеллекта.	ОР-7.2.2. ОР-7.4.1.	Задание 4
7.	Использование библиотек для работы с большими данными.	ОР-7.2.2. ОР-7.4.1.	Задание 2, Кейс 2

3. Типовые контрольные задания или иные материалы, необходимые для оценки образовательных результатов обучения

3.1. Типовые задания для проведения текущего контроля успеваемости по дисциплине

Задание 1. Составить программу на С# для перевода чисел из одной системы счисления в другую. Системы счисления произвольные. Для этого сначала реализуем алгоритм перевода чисел из системы счисления N в десятичную систему счисления, потом из десятичной переводим в систему счисления M.

Если $N = 10$ или $M = 10$ это частный случай, его надо проверить.

Составляем подробный отчёт, в котором должен быть код с подробными комментариями и скрины результатов выполнения программы во всех нужных случаях, граничных и обычных. Отчёт прикрепляем в Moodle, лучше в формате pdf.

Задание 2. Реализовать алгоритм Ханойские башни для любого N.

Прикрепляем подробный отчёт, в котором должен быть код с подробными комментариями и скрины результатов выполнения программы, здесь тоже подробно и с комментариями

Задание 3. Дано арифметическое выражение в виде строки. Найти его значение.

Прикрепляем подробный отчёт, в котором должен быть код с подробными комментариями и скрины результатов выполнения программы во всех нужных случаях, показать, как вычисляются выражения с одинарным минусом, скобками, ошибками.

Задание 4. Создать небольшой телеграм-бот для указанной рассылки, используя язык программирования Python. Подобрать нужные библиотеки. Прикрепляем подробный отчёт, в котором должен быть код с подробными комментариями и ссылка на разработанный бот.

Задание 5. Междисциплинарная, задача из теории графов. Реализовать алгоритм раскраски графа для графов большой размерности. Подобрать нужные библиотеки. Прикрепляем подробный отчёт, в котором должно быть обоснование выбранных средств реализации, код с подробными комментариями и скрины результатов.

Кейс 1. Междисциплинарный, задача из алгебры. Решить задачу в небольших группах, затем обсудить представленные программы.

Найти НОД($f(x), g(x)$) в конечном поле $Z(p)$. Задачу можно усложнить, взяв поле $Z(p^k)$.

Решение кейса должно быть представлено группе и преподавателю в виде отчёта и доклада с презентацией, возможно взаимное оценивание работы групп.

Кейс 2. Междисциплинарный, статистика. Сбор статистических данных из сети Интернет и исследование их характеристик.

Выбрать тему для исследования, решить задачу и представить результат.

Пример темы: построить корреляцию между продолжительностью жизни и отношения к курению.

Решение кейса должно быть представлено группе и преподавателю в виде отчёта и доклада с презентацией, возможно взаимное оценивание работы групп.

3.2. Типовые задания для проведения промежуточной аттестации по дисциплине

Тест по основам C#

1. Объект является
 - экземпляром функции
 - экземпляром класса
 - экземпляром структуры
 - экземпляром делегата
 - экземпляром события
2. Типы по значению хранятся
 - В управляемой куче
 - В стеке
 - В неуправляемой куче
3. Ссылочные типы хранятся
 - В управляемой куче
 - В стеке
 - В неуправляемой куче
4. К типам по значению в C# относятся
 - классы
 - структуры
 - делегаты
 - перечисления
 - встроенные (базовые) типы

- массивы
- 5. К ссылочным типам в C# относятся
 - классы
 - структуры
 - делегаты
 - перечисления
 - встроенные (базовые) типы
 - массивы
- 6. В C# к членам структуры по умолчанию устанавливается спецификация доступа
 - public
 - private
 - protected
 - internal
- 7. Статические методы класса могут обращаться
 - к не статическим методам класса
 - к статическим полям
 - к любым полям класса
 - к статическим методам класса
- 8. В C# к членам класса по умолчанию устанавливается спецификация доступа
 - private
 - public
 - protected
 - internal
- 9. В C# к членам интерфейса по умолчанию устанавливается спецификация доступа
 - private
 - public
 - protected
 - internal
- 10. Поля структуры можно инициализировать
 - При объявлении
 - В конструкторе с аргументами
 - В конструкторе без аргументов
- 11. Допускается не инициализировать перед вызовом функции параметр с модификатором
 - ref
 - out
 - static
- 12. Для передачи функции переменного количества параметров используется ключевое слово:
 - out
 - params
 - refg
- 13. Для передачи функции параметров по ссылке необходимо использовать ключевое слово
 - params
 - out
 - static
 - ref
- 14. Перегруженные функции отличаются
 - Типом возвращаемого значения.
 - Списком принимаемых значений.
 - И тем и другим.
 - Именем
- 15. Вызов статической функции класса осуществляется с использованием

- Ссылки на объект
- Имени класса
- Имени структуры
- 16. Конструктор имеет тип возвращаемого значения
 - void
 - не имеет
 - int
- 17. Конструктор по умолчанию может быть объявлен в
 - Структуре
 - Классе
 - Интерфейсе
- 18. Конструктор с аргументами может быть объявлен
 - Без спецификации доступа
 - Со спецификацией public
 - Со спецификацией private
 - Со спецификацией protected
- 19. Конструкторов по умолчанию в структуре может быть объявлено
 - один
 - ни одного
 - сколь угодно много
- 20. Константы инициализируются
 - При объявлении
 - С помощью конструктора по умолчанию
 - С помощью конструктора с аргументами
- 21. Конструктор может вызвать конструктор
 - Производного класса
 - Базового класса
 - Своего класса
- 22. Для вызова конструктора базового класса необходимо использовать
 - имя базового класса
 - ключевое слово base
 - ключевое слово this

4. Методические материалы, определяющие процедуры оценивания образовательных результатов обучения

4.1. Методические материалы для оценки текущего контроля успеваемости по дисциплине.

Задание 1-5

Критерии	0 баллов	1 балл	2 балла
Корректность постановки задачи (если есть)	Задача поставлена не корректно, непонятно что должна делать программа.	Задача сформулирована корректно, но с ошибками.	Задача сформулирована корректно.
Правильность выбора средств реализации	Средства реализации выбраны неправильно, задачу реализовать тяжело.	Средства реализации выбраны правильно, но существуют более подходящие средства.	Средства реализации выбраны правильно, задачу легко реализовать.

Правильность алгоритма	Алгоритм реализован неправильно, задача не решена	Алгоритм реализован так, что задача решена частично.	Алгоритм реализован правильно, задача решена
Правильность работы программы	Облачные технологии не использованы	Программа использует облачные технологии в недостаточном объёме.	Программа работает с облачными технологиями
Адекватность отчёта	Отчёт не сделан	Отчёт не даёт представления о решении задачи.	Отчёт даёт представление о решении задачи.

Кейсы 1, 2

Критерии	0 баллов	1 балл	2 балла
Корректность постановки задачи (если есть)	Задача поставлена не корректно, непонятно что должна делать программа.	Задача сформулирована корректно, но с ошибками.	Задача сформулирована корректно.
Правильность выбора средств реализации	Средства реализации выбраны неправильно, задачу реализовать тяжело.	Средства реализации выбраны правильно, но существуют более подходящие средства.	Средства реализации выбраны правильно, задачу легко реализовать.
Правильность алгоритма	Алгоритм реализован неправильно, задача не решена	Алгоритм реализован так, что задача решена частично.	Алгоритм реализован правильно, задача решена
Правильность работы программы	Облачные технологии не использованы	Программа использует облачные технологии в недостаточном объёме.	Программа работает с облачными технологиями
Адекватность отчёта	Отчёт не сделан	Отчёт не даёт представления о решении задачи.	Отчёт даёт представление о решении задачи.
Качество оформления презентации	Небрежное оформления презентации.	Оформление недостаточно продумано и аккуратно	Оформление продумано и аккуратно
Умение отвечать на вопросы	Студент не ответил на вопросы	Студент ответил только на часть вопросов	Студент ответил на все вопросы
Умение задавать вопросы	Студент не задаёт вопросы другим выступающим	Студент задал 2-3 вопроса другим выступающим	Студент активно задаёт вопросы другим выступающим

4.2. Методические материалы для проведения промежуточной аттестации по дисциплине.

Тест оценивается следующим образом:

Отлично – от 97% до 100% правильных ответов

Хорошо – от 90% до 96% правильных ответов

Удовлетворительно – от 70% до 89% правильных ответов

Неудовлетворительно – менее 69% правильных ответов

Зачёт с оценкой выставляется по результатам проверки лабораторных работ и кейсов (70%), и оценки за тест (30%). При этом за задания и кейсы максимально можно набрать 14 баллов. Оценка выставляется следующим образом:

Отлично – от 12 до 14 баллов

Хорошо – от 9 до 11 баллов

Удовлетворительно – от 6 до 8 баллов

Неудовлетворительно – менее 5 баллов