Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра автоматизированных систем управления
Кафедра автоматики
Кафедра вычислительной техники
Кафедра защиты информации

Паспорт
расчетно-графической работы

по дисциплине «Программирование», 2 семестр

Методика оценки

Выполнение расчетно-графической работы (далее - РГР) является формой текущей аттестации (контроля) по дисциплине, предусмотренной учебным планом.

Цель РГР: освоение разработки и программной реализации алгоритмов на C++17 и Python, работы с типами данных, памятью, контейнерами, потоками, сериализацией и учебными криптографическими вычислениями; формирование навыков выбора инструментов, отладки и безопасного программирования по индикаторам ОПК-2.1, ОПК-2.2, ОПК-7.1, ОПК-7.2 и ОПК-7.3.

Обязательным элементом РГР является решение профессиональных задач разработки и проверки программ в соответствии с выбранной темой.

Номер задания из перечня п. 4 определяется преподавателем. Для каждого студента задаются индивидуальные исходные данные и параметры; тема и исходные данные фиксируются в задании на РГР.

РГР выполняется индивидуально.

Количество тем и индивидуальных наборов исходных данных обеспечивает каждого обучающегося индивидуальным заданием РГР.

Замена задания РГР осуществляется по согласованию с преподавателем из числа резервных (не занятых) заданий.

Преподаватель осуществляет руководство по выполнению задания, оказывает консультационную помощь и принимает отчет по РГР.

По результатам выполнения РГР выполняется отчет, который состоит из следующих частей:

Титульный лист.

Введение: цель, постановка задания, исходные данные и используемые источники.

Теоретическая часть: описание выбранных методов, алгоритмов и программных средств.

Практическая часть: структуры данных, алгоритмы и реализация, тестовые наборы, результаты проверки и оценка сложности или производительности.

Выводы или заключение.

Список литературы.

Приложения (при необходимости).

Требования к оформлению:

Объем РГР до 50 страниц машинописного текста формата А4. Шрифт Times New Roman, 12. Формулы набираются в редакторе Math Type. Размещение сканированных формул не допускается. Нумерация страниц сквозная, в нижней части листа по центру арабскими цифрами. Работа должна быть отредактирована, не содержать орфографических, синтаксических и стилевых ошибок.

Отчет в установленные сроки сдается на кафедру для проверки. Преподаватель оценивает качество работы, отмечает положительные стороны и недостатки работы и определяет, допускается ли она к защите. При необходимости преподаватель возвращает РГР студенту для доработки и устанавливает сроки повторного предоставления для проверки. До защиты работы студентом должны быть сделаны необходимые исправления и дополнения по всем замечаниям преподавателя.

При положительном результате оценивания РГР студент её распечатывает, передает на кафедру и защищает до сессии в назначенное преподавателем время.

Защита РГР состоит в индивидуальном устном собеседовании студента с преподавателем. В процессе защиты выявляется уровень знаний студента, степень его самостоятельности при выполнении работы. По результатам защиты студенту выставляется оценка в соответствии с критериями, приведенными в п. 2 настоящего Паспорта.

Критерии оценки результатов обучения, соотнесенных с уровнями освоения индикаторов достижения компетенций

Общие правила выставления оценок текущей аттестации определяются балльно-рейтинговой системой, установленной локальным актом НГТУ.

РГР считается выполненной **на продвинутом** уровне, если все необходимые алгоритмы реализованы самостоятельно и без ошибок; все разделы РГР выполнены правильно и в полном объеме; работа оформлена в соответствии с требованиями; сдана преподавателю в указанные сроки и не возвращалась для доработки; даны полные и развернутые выводы и рекомендации; на защите студентом даны уверенные и аргументированные ответы. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций не содержит пробелов. Закрепленные за РГР компетенции сформированы на продвинутом уровне. Оценка составляет *от 18 до 20 баллов.*

РГР считается выполненной **на базовом** уровне, если все необходимые алгоритмы реализованы самостоятельно и без существенных ошибок; все разделы РГР выполнены правильно, но есть замечания к полноте предоставления информации; работа оформлена в соответствии с требованиями; сдана преподавателю в указанные сроки и однократно возвращалась студенту для незначительной доработки; в заключении даны выводы и рекомендации; на защите студентом допущены непринципиальные ошибки. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций содержит несущественные пробелы. Закрепленные за РГР компетенции сформированы на базовом уровне. Оценка составляет *от 14 до 17 баллов.*

РГР считается выполненной **на пороговом** уровне, если все необходимые алгоритмы реализованы самостоятельно, но с ошибками, часть из которых носит принципиальный характер; есть замечания к полноте предоставления информации; работа оформлена в соответствии с требованиями; сдана преподавателю, но неоднократно возвращалась студенту для доработки; в заключении даны краткие выводы; защита РГР вызывает у студента серьезные затруднения. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций содержит существенные пробелы. Закрепленные за РГР компетенции сформированы на пороговом уровне. Оценка составляет *от 10 до 13 баллов.*

РГР считается **не выполненной** (ниже порогового уровня), если алгоритмы реализованы с серьезными ошибками; есть замечания к полноте предоставления информации и оформлению; РГР была сдана преподавателю, но неоднократно возвращалась студенту для доработки, что не привело к улучшению ее качества; РГР не допущена до защиты, что свидетельствует о неудовлетворительном уровне достигнутых студентом результатов. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций содержит множественные существенные пробелы. Закрепленные за РГР компетенции не сформированы. Оценка составляет менее 10 баллов.

Шкала оценки

В общей оценке по дисциплине баллы за РГР учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

РГР как форма текущей аттестации (контроля) по дисциплине считается успешно выполненной, если сумма полученных баллов составляет от 10 до 20 баллов включительно.

Примерный перечень тем РГР

1. Обработка числовых данных на C++17 и Python. Реализовать чтение последовательности, расчёт статистик и фильтрацию; сравнить типы данных, обработку ошибок и результаты двух реализаций.
2. Исследование сортировки и поиска. Реализовать два метода сортировки, линейный и двоичный поиск; сравнить число операций и время на разных наборах данных, обосновать сложность.
3. Безопасная обработка динамического массива. Реализовать ввод, вставку, удаление и изменение размера; проверить границы, освобождение памяти и поведение при ошибочном вводе.
4. Обработка многомерных данных. Реализовать операции над матрицами с динамическим выделением памяти и с vector<vector>; сравнить адресацию и организацию хранения.
5. Рекурсивная и итерационная обработка данных. Разработать обе реализации согласованного алгоритма, проверить базовые случаи, сравнить глубину стека и трудоёмкость.
6. Каталог записей на основе структур. Реализовать хранение записей, сортировку по полям, поиск и фильтрацию; организовать многофайловый проект и файловый ввод-вывод.
7. Частотный анализ текста с map и set. Подсчитать частоты слов, сформировать множество уникальных элементов и выполнить поиск; проверить пустые файлы и используемую кодировку.

8. Сравнение способов хранения множества ключей. Реализовать поиск и удаление в vector, map и set; исследовать трудоёмкость, дубликаты и критерии выбора контейнера.
9. Обработка данных алгоритмами STL. Использовать algorithm и лямбда-выражения для отбора, преобразования и сортировки записей; сопоставить с собственной реализацией циклов.
10. Потоковое преобразование табличного текстового файла. Использовать файловые и строковые потоки, проверять формат записей, выделять ошибочные строки и сохранять корректный результат.
11. Исследование буферизации вывода. Сравнить режимы вывода, связность потоков и частоту сброса буфера; провести повторные замеры при неизменных данных и проверить правильность результата.
12. Модуль целочисленной арифметики. Реализовать НОД, НОК, проверку делимости и поиск простых чисел; проверить граничные значения и избежать переполнения.
13. Вычисления по модулю. Реализовать быстрое возведение в степень, функцию Эйлера и проверку применимости теорем Эйлера и Ферма на учебных данных.
14. Решение линейных сравнений. Реализовать расширенный алгоритм Евклида и поиск решений сравнения; проверять условия существования и полноту множества решений.
15. Исследование двучленных и показательных сравнений. На небольших учебных модулях реализовать поиск решений и таблицу индексов; сопоставить результаты с теоретическими условиями разрешимости.
16. Учебное сравнение симметричного и асимметричного шифрования. Продемонстрировать шифрование и восстановление тестовых данных согласованными библиотечными средствами; описать роль ключей и ограничения учебной модели.
17. Бинарное сохранение структурированных данных. Разработать явно описанный формат, чтение и запись записей; проверить длины, порядок байтов, обрезанные файлы и восстановление данных.
18. Обмен данными в JSON. Реализовать сериализацию и десериализацию набора записей на C++17 и Python; проверить обязательные поля, типы, настройки преобразования и совпадение результатов.
19. Обмен данными в XML. Реализовать чтение и запись структурированных данных, согласовать элементы и атрибуты, проверить структуру документа и отклонять некорректный ввод.
20. Сравнение дружественных функций и get/set. Реализовать две версии доступа к данным пользовательского типа; показать влияние на инкапсуляцию, корректность и тестирование в многофайловом проекте.