

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«АРКТИЧЕСКИЙ ГОСУДАРСТВЕННЫЙ АГРОТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО Арктический ГАТУ)
Колледж технологий и управления

Регистрационный
№ 24-17/25

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Дисциплина **ОП.06 Основы алгоритмизации и программирования**

Специальность **09.02.11 Разработка и управление программным обеспечением**

Квалификация **Программист**

Уровень **ППССЗ базовая**

Срок освоения **ППССЗ 3 г 10 мес**

Форма обучения **очная**

Общая трудоемкость **216 ч**

Якутск 2026

Рабочая программа учебной дисциплины разработана в соответствии с:

- Федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением, утвержденным приказом Министерства образования и науки Российской Федерации от 24.02.2025 № 138.
- Учебным планом специальности 09.02.11 Разработка и управление программным обеспечением, одобренным Ученым советом ФГБОУ ВО Арктический ГАТУ от 02.04.2026 г. № 56.

Разработчик(и) РПД Федоров Павел Иванович – преподаватель

Председатель цикловой комиссии ГиЕД _____ /Васильева Е.К./
подпись фамилия, имя, отчество

Протокол заседания цикловой комиссии ГиЕД №10 от «13» марта 2026 г

Директор КТиУ _____ /Яковлева Н.М./
подпись фамилия, имя, отчество

«20» апреля 2026 г.

Председатель МК КТиУ _____ /Ваганова В.Г./
подпись фамилия, имя, отчество

Протокол заседания МК КТиУ № 3 от «20» марта 2026 г

СОДЕРЖАНИЕ

№	Наименование раздела	Стр.
1	Общая характеристика рабочей программы учебной дисциплины	2
2	Структура и содержание учебной дисциплины	4
3	Условия реализации учебной дисциплины	9
4	Контроль и оценка результатов освоения учебной дисциплины	10

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

09.02.11 Разработка и управление программным обеспечением

1.1. Область применения программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности СПО 09.02.11 Разработка и управление программным обеспечением.

1.2. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена

Учебная дисциплина ОП.06 Основы алгоритмизации и программирования относится к общепрофессиональному циклу.

Освоение дисциплины способствует формированию компетенций:

ОК 01 – выбирать способы решения задач (алгоритмы, структуры данных, технологии).

ОК 02 – искать и анализировать информацию (документация, форумы, ЭБС).

ОК 03 – планировать развитие (изучение новых языков/инструментов, GitHub).

ОК 04 – работать в команде (групповые проекты, парное программирование).

ОК 05 – общаться устно и письменно (отчёты, защита проектов, комментарии в коде).

ОК 06 – соблюдать этику и закон (лицензии ПО, антиплагиат, авторское право).

ОК 07 – экономить ресурсы (оптимизация кода, энергоэффективные алгоритмы).

ОК 08 – использовать физкультуру для здоровья (перерывы, эргономика рабочего места).

ОК 09 – работать с документацией на русском и иностранном языках (англоязычная документация, Stack Overflow, MSDN).

ПК 1.1 – анализировать требования к ПО (техзадание, выбор инструментов).

ПК 2.2 – выполнять интеграцию модулей (сборка проекта, взаимодействие классов/функций).

ПК 2.4 – разрабатывать тесты и сценарии (отладка, тестовые наборы, проверка граничных случаев).

1.3. Цели и задачи учебной дисциплины – требования к результатам освоения

Цель дисциплины: обеспечение обучающихся теоретическими знаниями и умениями, практическими навыками, необходимыми для разработки алгоритмов и программных продуктов на языках высокого уровня с использованием современных технологий программирования.

Задача дисциплины: изучение основ алгоритмизации, базовых конструкций языков программирования, принципов модульного и объектно-ориентированного программирования, а также приобретение навыков создания и тестирования приложений.

В результате освоения учебной дисциплины обучающийся должен уметь:

У.1. Разрабатывать алгоритмы решения задач и реализовывать их на языке программирования.

У.2. Использовать языки программирования, типы данных, операторы и функции для создания программ.

У.3. Применять принципы модульного и объектно-ориентированного программирования.

У.4. Выполнять отладку и тестирование программного кода.

У.5. Работать с файловой системой, осуществлять ввод/вывод данных.

В результате освоения учебной дисциплины обучающийся должен знать:

З.1. Этапы решения задач на компьютере, свойства и виды алгоритмов, способы их записи.

З.2. Базовые конструкции языков программирования (переменные, типы данных, операторы, циклы, массивы).

З.3. Принципы структурного, модульного и объектно-ориентированного

программирования.

3.4. Основные принципы визуально-событийного управления, этапы разработки приложений.

3.5. Методы тестирования и отладки программного обеспечения.

1.4. Рекомендуемое количество часов на освоение программы учебной дисциплины

Максимальной учебной нагрузки обучающегося **216** часов, в том числе:

- обязательной аудиторной учебной нагрузки обучающегося **186** часов;
- самостоятельной работы обучающегося **12** часов;
- промежуточная аттестация – **18** часов (из них 5 часов – экзамен).

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	216
Обязательная аудиторная учебная нагрузка (всего)	186
в том числе:	
лекции	84
практические занятия	102
Самостоятельная работа студента (всего)	12
Промежуточная аттестация (из них 5 часов – экзамен)	18
Итоговая аттестация в форме	экзамен

2.2. Тематический план и содержание учебной дисциплины ОП.2 Основы алгоритмизации и программирования

<i>Наименование разделов и тем</i>	<i>Содержание учебного материала, практические занятия, самостоятельная работа обучающихся</i>	<i>Объем часов</i>	<i>В том числе часы по практической подготовке</i>	<i>Уровень освоения</i>
Раздел 1. Введение в программирование		50	30	
Тема 1.1. Основы алгоритмизации, языки и системы программирования	<i>Содержание учебного материала</i>			
	Лекция №1. Алгоритм и его свойства. Виды алгоритмов. Способы записи алгоритмов. Сложность алгоритмов.	2		1
	Лекция №2. Эволюция и классификация языков программирования. Среда программирования. Компиляторы и интерпретаторы.	2		1
	Лекция №3. Жизненный цикл программы. Программа. Программный продукт и его характеристики.	2		1
	Лекция №4. Основные элементы языка. Типы данных. Простые, производные и структурированные типы данных.	2		2
	Лекция №5. Операторы языка. Ввод/вывод данных. Обработка исключений. Операторы разветвляющихся программ.	2		2
	Лекция №6. Циклические программы. Одномерные массивы. Двумерные массивы. Работа с массивами. Сортировка массивов.	2		2
	Лекция №7. Строки. Коллекции. Файлы. Доступ к файлам. Виды файлов. Считывание и запись в файл.	2		2
	<i>Практические занятия:</i>			
	Практическое занятие №1. Линейные программы.	2	2	2
	Практическое занятие №2. Составление программ разветвляющейся структуры.	2	2	2
	Практическое занятие №3. Циклические программы.	2	2	2
	Практическое занятие №4. Одномерные массивы.	2	2	2
	Практическое занятие №5. Двумерные массивы.	2	2	2
	Практическое занятие №6. Обработка массивов.	2	2	2
	Практическое занятие №7. Сортировка массивов.	4	4	2
	Практическое занятие №8. Символы и строки. Обработка строк.	2	2	2
	Практическое занятие №9. Использование коллекций.	2	2	2
	Практическое занятие №10. Работа с файлами.	4	4	2

	Практическое занятие №11. Работа с каталогами и файлами.	2	2	2
	<i>Самостоятельная работа:</i> проработка конспектов лекций, подготовка к практическим занятиям.	4		3
Раздел 2. Технологии программирования		76	36	
Тема 2.1. Модульное программирование	<i>Содержание учебного материала</i>			
	Лекция №8. Модульное программирование. Локальные и глобальные переменные.	2		1
	Лекция №9. Подпрограммы. Модификаторы. Передача данных в подпрограммы.	2		2
	Лекция №10. Рекурсия. Разработка рекурсивных подпрограмм.	2		2
	<i>Практические занятия:</i>			
	Практическое занятие №12. Использование подпрограмм.	4	4	2
	Практическое занятие №13. Рекурсия.	4	4	2
	Практическое занятие №14. Создание модулей.	2	2	2
Тема 2.2. Основные принципы объектно-ориентированного программирования	<i>Содержание учебного материала</i>			
	Лекция №11. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	2		1
	Лекция №12. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.	2		2
	Лекция №13. Абстрактные классы и интерфейсы.	2		2
	<i>Практические занятия:</i>			
	Практическое занятие №15. Работа с классами. Создание конструкторов.	2	2	2
	Практическое занятие №16. Применение свойств.	2	2	2
	Практическое занятие №17. Наследование.	4	4	2
	Практическое занятие №18. Полиморфизм.	4	4	2
	Практическое занятие №19. Применение принципов ООП.	4	4	2
	<i>Самостоятельная работа:</i> проработка конспектов лекций, подготовка к практическим занятиям.	4		3
Раздел 3. Разработка приложений		72	36	
Тема 3.1. Этапы разработки	<i>Содержание учебного материала</i>			
	Лекция №14. Визуально-событийно управляемое программирование.	2		2

приложений	Лекция №15. Разработка приложения. Проектирование ООП приложения.	2		2
	Лекция №16. Создание интерфейса пользователя.	2		2
	Лекция №17. Компоненты для работы с текстом, кнопки, переключатели.	2		2
	Лекция №18. Компоненты для отображения таблиц, дат и времени.	2		2
	Лекция №19. Тестирование, отладка приложения.	2		2
	Лекция №20. Оптимизация программы.	2		2
	<i>Практические занятия:</i>			
	Практическое занятие №20. Создание проекта с использованием компонентов для работы с текстом.	2	2	2
	Практическое занятие №21. Создание проекта с использованием кнопочных компонентов.	2	2	2
	Практическое занятие №22. Создание проекта с использованием переключателей.	2	2	2
	Практическое занятие №23. Создание проекта с использованием компонентов для отображения таблиц.	2	2	2
	Практическое занятие №24. Создание проекта с использованием компонентов для отображения дат и времени.	2	2	2
	Практическое занятие №25. Разработка интерфейса приложения.	6	6	2
	Практическое занятие №26. Тестирование и отладка приложения.	4	4	2
	Практическое занятие №27. Оптимизация программы.	4	4	2
	Практическое занятие №28. Разработка полноценного приложения (проект).	8	8	2
	<i>Самостоятельная работа:</i> оформление отчетов по практическим работам, подготовка к экзамену.	4		3
Промежуточная аттестация	Экзамен (5 часов – выполнение комплексного практического задания, 13 часов – консультации и резерв)	18		
Всего:		216	102	

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

№ п/п	Наименование дисциплины (модуля), практик в соответствии с учебным планом	Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы
1	ОП.2 Основы алгоритмизации и программирования	Ауд. № 2.114 (78,0 м²). Мультимедийный зал научной библиотеки для самостоятельной работы с выходом в сеть Интернет	Оборудование: Системный блок Core quad q6600, 4GB RAM, 160GB HDD – 1 шт.; Монитор Benq G900WA – 1 шт.; Системный блок Deponeon core2duo e8300, 2GB RAM, 160GB HDD – 8 шт.; Монитор LG W1934S – 8 шт.; Тонкие клиенты Eltex TC-50 – 4 шт. Учебная мебель: Компьютерный стол – 15 шт., стол – 9 шт., стулья – 23 шт. Программное обеспечение: Бесплатная операционная система Calculate Linux; LIBREOFFICE (открытое лицензионное соглашение GNU General Public License).
2	ОП.2 Основы алгоритмизации и программирования	Ауд. № 2.313 (52,5 м²). Компьютерный класс информационных технологий в профессиональной деятельности	Оборудование: Автоматизированные рабочие места на 14 обучающихся (NL Intel(R) Core(TM) i3-10100/2*4Gb/GeForce RTX 3050/500Gb/Win10Pro/Office, монитор 23.8" AOC 24B2XHM2); Автоматизированное рабочее место преподавателя (NL Intel(R) Core(TM) i3-10100/2*4Gb/GeForce RTX 3050/500Gb/Win10Pro/Office, монитор 23.8" AOC 24B2XHM2) – 1 шт.; Проектор и экран – 1 комплект. Учебная мебель: Компьютерный стол – 16 шт., стул подъемно-поворотный – 1 шт., стулья со спинкой – 25 шт., шкаф для документов – 1 шт., доска трехэлементная для написания мелом и фломастером (3000x1000x20) – 1 шт.
3	ОП.2 Основы алгоритмизации и программирования	Ауд. № 2.406. Компьютерный класс. Лаборатория информационных ресурсов	Оборудование: Автоматизированные рабочие места обучающихся: системный блок (Rusco Core-i3-7100/2*4Gb/500Gb/Win10Pro/Office) – 15 шт., монитор (22" Benq GL2250) – 15 шт.; Автоматизированное рабочее место преподавателя: системный блок (Rusco Core-i3-7100/2*4Gb/500Gb/Win10Pro/Office) – 1 шт., монитор (22" Benq GL2250) – 1

			шт.; Интерактивная доска SMART Board 680; Проектор LG RL-JT40; МФУ HP LaserJet Pro MFP 127fn – 1 шт.; навесной экран, маркерная доска. Учебная мебель: Специализированная мебель для сервисного обслуживания ПК с заземлением и защитой от статического напряжения, компьютерный стол – 16 шт., стул подъемно-поворотный – 16 шт., стулья – 25 шт.
--	--	--	---

3.2. Информационное обеспечение обучения

Перечень учебных изданий, интернет-ресурсов, дополнительной литературы

Основные источники:

№	Авторы, составители	Заглавие	Издательство, год
1	Трофимов, В. В.	Основы алгоритмизации и программирования : учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская. — 4-е изд. — Москва : Издательство Юрайт, 2025. — 108 с. — (Профессиональное образование). — ISBN 978-5-534-20429-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/56386	ЭЮС Юрайт
2	Кудрина, Е. В.	Основы алгоритмизации и программирования на языке С# : учебник для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2026. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/587050	ЭБС Юрайт
3	Якимов, С. П	Алгоритмизация и программирование: учебник для среднего профессионального образования / С. П. Якимов. — Москва : Издательство Юрайт, 2026. — 342 с. — (Профессиональное образование). — ISBN 978-5-534-19661-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/589849	ЭБС Юрайт

Дополнительная литература

№	Авторы, составители	Заглавие	Издательство, год
1	А.Д. Хомоненко	Алгоритмизация и программирование. Задачи повышенной сложности	СПб.: БХВ-Петербург, 2024

2	Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес	Паттерны объектно-ориентированного проектирования	СПб.: Питер, 2024
3	С.В. Куликов	Тестирование программного обеспечения. Базовый курс	Москва: ДМК Пресс, 2025

Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

№	Наименование
1	Электронно-библиотечная система «Юрайт» для СПО — Режим доступа: https://urait.ru
2	Электронно-библиотечная система «Лань» — Режим доступа: https://e.lanbook.com
3	Национальный открытый университет «ИНТУИТ» — Курсы по программированию для СПО — Режим доступа: https://www.intuit.ru
4	Профессиональный форум программистов «Stack Overflow» — Режим доступа: https://stackoverflow.com
5	Официальная документация Microsoft C# — Режим доступа: https://docs.microsoft.com/dotnet/csharp
6	Портал «КиберФорум» (разделы C#, Python, ООП) — Режим доступа: https://www.cyberforum.ru
7	Ресурс для подготовки к демонстрационному экзамену по компетенции «Программные решения для бизнеса» — Режим доступа: https://esat.worldskills.ru
8	База знаний Git и GitHub — Режим доступа: https://docs.github.com
9	Stepik — онлайн-курсы по программированию (ООП, алгоритмы) — Режим доступа: https://stepik.org
10	Методические материалы по разработке ПО (портал «Открытое образование») — Режим доступа: https://openedu.ru
11	Видеолекции по технологии разработки ПО (канал «Hexlet» на YouTube)
12	Система дистанционного обучения Moodle АГАТУ — Режим доступа: http://sdo.agatu.ru
13	Научная электронная библиотека «КиберЛенинка» — Режим доступа: https://cyberleninka.ru

Комплект лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства

№	Наименование
1	Операционная система Calculate Linux (свободно распространяемое)
2	LIBREOFFICE (открытое лицензионное соглашение GNU General Public License)
3	Операционная система Windows 10 Pro (лицензионное)
4	Microsoft Office (лицензионное)
5	Среда разработки (IDE) с настраиваемым интерфейсом (например, Visual Studio Community, VS Code)
6	Система дистанционного обучения Moodle

Перечень профессиональных баз данных и информационных справочных систем:

№	Наименование
1	Справочная правовая система «КонсультантПлюс» — Режим доступа: http://www.consultant.ru
2	Профессиональный стандарт «Программист» (Приказ Минтруда) — доступ через ИПС «Гарант»
3	Информационная система «Единое окно доступа к образовательным ресурсам» —

	Режим доступа: http://window.edu.ru
4	Электронный фонд правовой и нормативно-технической документации — Режим доступа: http://docs.cntd.ru

3.3. Условия реализации учебной дисциплины для студентов-инвалидов и лиц с ограниченными возможностями здоровья

3.3.1. Образовательные технологии

С целью оказания помощи в обучении студентов-инвалидов и лиц с ОВЗ применяются образовательные технологии с использованием универсальных, специальных информационных и коммуникационных средств.

Для основных видов учебной работы применяются:

Контактная работа:

- *лекции* – проблемная лекция, лекция-дискуссия, лекция-диалог, лекция-консультация, лекция с применением дистанционных технологий и привлечением возможностей Интернета;
- *практические (семинарские) занятия* – практические задания;
- *групповые консультации* – опрос, работа с лекционным и дополнительным материалом;
- *индивидуальная работа с преподавателем* – индивидуальная консультация, работа с лекционным и дополнительным материалом, беседа, морально-эмоциональная поддержка и стимулирование, дистанционные технологии.

Формы самостоятельной работы устанавливаются с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге или на компьютере).

В качестве самостоятельной подготовки в обучении используется система дистанционного обучения Moodle.

Самостоятельная работа:

- работа с книгой и другими источниками информации, план-конспекты;
- творческие самостоятельные работы;
- дистанционные технологии.

При необходимости обучающимся предоставляется дополнительное время для консультаций и выполнения заданий.

3.3.2. Специальное материально-техническое и учебно-методическое обеспечение

При обучении по дисциплине используется система, поддерживающая дистанционное образование – «Moodle», ориентированная на организацию дистанционных курсов, а также на организацию взаимодействия между преподавателем и обучающимися посредством интерактивных обучающих элементов курса.

Для обучающихся лиц с нарушением зрения предоставляются:

- видеоувеличитель-монокюль для просмотра Levenhuk Wise 8x25;
- электронный ручной видеоувеличитель видео оптик “wu-tv”;
- возможно также использование собственных увеличивающих устройств;
- версия сайта академии <http://www.agatu.ru/> для слабовидящих;
- программное обеспечение с поддержкой экранного доступа (NVDA, JAWS);
- учебные пособия, методические указания в форме электронного документа с возможностью увеличения шрифта;
- IDE (среды разработки) с настраиваемым интерфейсом (темная тема, крупный шрифт).

Для обучающихся лиц с нарушением слуха предоставляются:

- аудитории со звукоусиливающей аппаратурой (колонки, микрофон);
- компьютерная техника в оборудованных классах;
- учебные аудитории с мультимедийной системой с проектором;
- аудитории с интерактивными досками;
- учебные пособия, методические указания в форме электронного документа;

- лекции в формате презентаций с текстовым сопровождением, видео с субтитрами;
- использование систем видеоконференцсвязи с функцией текстового чата.

Для обучающихся лиц с нарушениями опорно-двигательного аппарата предоставляются:

- система дистанционного обучения Moodle;
- учебные пособия, методические указания в форме электронного документа;
- возможность использования альтернативных устройств ввода информации (голосовой ввод, адаптированная клавиатура, трекбол);
- рабочее место, адаптированное для доступа и перемещения (регулируемая высота стола, специальное кресло);
- программное обеспечение с поддержкой голосового управления.

3.3.3. Контроль и оценка результатов освоения учебной дисциплины

Контроль результатов обучения осуществляется в процессе проведения практических занятий, выполнения индивидуальных самостоятельных работ.

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации инвалидов и лиц с ОВЗ имеются фонды оценочных средств в ИС «Тестирование».

Формы и сроки проведения рубежного контроля определяются с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.), и может проводиться в несколько этапов. При необходимости, предоставляется дополнительное время для подготовки ответов на зачете, аттестация проводится в несколько этапов (по частям), во время аттестации может присутствовать ассистент, аттестация прерывается для приема пищи, лекарств, во время аттестации используются специальные технические средства.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<i>Уметь</i>	
У.1. Разрабатывать алгоритмы решения задач и реализовывать их на языке программирования.	Оценка выполнения и защиты практических работ (№1-7, 13-15). Наблюдение за деятельностью обучающегося.
У.2. Использовать языки программирования, типы данных, операторы и функции для создания программ.	Оценка выполнения практических работ (№1-20). Тестирование по темам разделов 1 и 2. Проверка самостоятельной работы.
У.3. Применять принципы модульного и объектно-ориентированного программирования.	Оценка выполнения и защиты практических работ (№13-20). Экспертная оценка индивидуальных заданий.
У.4. Выполнять отладку и тестирование программного кода.	Оценка выполнения практических работ (№26-28). Наблюдение за процессом отладки.
У.5. Работать с файловой системой, осуществлять ввод/вывод данных.	Оценка выполнения практических работ (№10-12). Проверка отчетов.
<i>Знать</i>	
3.1. Этапы решения задач на компьютере, свойства и виды алгоритмов, способы их записи.	Устный и письменный опрос (лекции №1-2). Тестирование. Проверка конспектов.
3.2. Базовые конструкции языков программирования.	Тестирование. Оценка выполнения практических работ (№1-8). Контрольная работа по разделу 1.
3.3. Принципы структурного, модульного и объектно-ориентированного программирования.	Дифференцированный зачет по разделу 2. Оценка выполнения практических работ (№13-20).
3.4. Основные принципы визуально-событийного управления, этапы разработки приложений.	Оценка выполнения практических работ (№21-27). Защита индивидуального проекта.
3.5. Методы тестирования и отладки программного обеспечения.	Устный опрос (лекция №13). Оценка выполнения практической работы №28.
Итоговый контроль:	Экзамен (оценка выполнения комплексного практического задания и ответов на теоретические вопросы).

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Арктический государственный агротехнологический университет»
Колледж технологий и управления

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по учебной дисциплине
ОП.06 Основы алгоритмизации и программирования
для обучающихся по специальности
09.02.11 Разработка и управление программным обеспечением

Якутск 2026 г

Фонд оценочных средств учебной дисциплины разработан в соответствии с:

- Федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением, утвержденным приказом Министерства образования и науки Российской Федерации от 24.02.2025 № 138.
- Учебным планом специальности 09.02.11 Разработка и управление программным обеспечением, одобренным Ученым советом ФГБОУ ВО Арктический ГАТУ от 02.04.2026 г. № 56.

Разработчик(и) ФОС Федоров Павел Иванович – преподаватель

Фонд оценочных средств учебной дисциплины ОП.06 Основы алгоритмизации и программирования одобрен Цикловой комиссией гуманитарных и естественных дисциплин от 13 марта 2026 г, протокол №10.

Председатель ЦК ГиЕД _____


подпись

/Васильева Е.К./
фамилия, имя, отчество

Фонд оценочных средств учебной дисциплины рассмотрен и рекомендован к использованию в учебном процессе на заседании методической комиссии Колледжа технологий и управления по специальности 09.02.11 Разработка и управление программным обеспечением.

Председатель методической комиссии КТиУ _____


подпись

/Ваганова В.Г./
фамилия, имя, отчество

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
по дисциплине Основы алгоритмизации и программирования
по специальности 09.02.11 Разработка и управление программным обеспечением

Таблица 1

Результаты обучения (освоенные умения, усвоенные знания)	Формируемые компетенции	Наименование темы	Уровень освоения Темы	Наименование контрольно-оценочного средства	
				Текущий контроль	Промежуточная аттестация
Уметь:					
У.1. Разрабатывать алгоритмы решения задач и реализовывать их на языке программирования.	ОК 01, ОК 02, ПК 1.1	Тема 1.1, Тема 2.1	2,3	Практические занятия №1-7, 12-13; Наблюдение	Экзаменационные задания
У.2. Использовать языки программирования, типы данных, операторы и функции для создания программ.	ОК 02, ПК 2.4	Тема 1.1	2,3	Практические занятия №1-11; Тестирование	Экзаменационные задания
У.3. Применять принципы модульного и объектно-ориентированного программирования.	ОК 01, ОК 02, ПК 2.2	Тема 2.1, Тема 2.2	2,3	Практические занятия №12-19; Индивидуальные задания	Экзаменационные задания
У.4. Выполнять отладку и тестирование программного кода.	ПК 2.4	Тема 3.1	2,3	Практические занятия №26-28; Наблюдение	Экзаменационные задания
У.5. Работать с файловой системой, осуществлять ввод/вывод данных.	ПК 2.2	Тема 1.1	2	Практические занятия №10-11; Проверка отчетов	Экзаменационные задания
Знать:					

3.1. Этапы решения задач на компьютере, свойства и виды алгоритмов, способы их записи.	ОК 01, ОК 02	Тема 1.1	1	Устный опрос, Тестирование	Экзаменационные вопросы
3.2. Базовые конструкции языков программирования (переменные, типы данных, операторы, циклы, массивы).	ОК 02, ПК 1.1	Тема 1.1	1,2	Тестирование, Контрольная работа по разделу 1	Экзаменационные вопросы
3.3. Принципы структурного, модульного и объектно-ориентированного программирования.	ОК 01, ПК 2.2	Тема 2.1, Тема 2.2	1,2	Устный опрос, Практические занятия	Экзаменационные вопросы
3.4. Основные принципы визуально-событийного управления, этапы разработки приложений.	ОК 02, ПК 1.1	Тема 3.1	1,2	Защита индивидуального проекта	Экзаменационные вопросы
3.5. Методы тестирования и отладки программного обеспечения.	ПК 2.4	Тема 3.1	1	Устный опрос	Экзаменационные вопросы

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

В результате аттестации по учебной дисциплине осуществляется комплексная проверка следующих умений и знаний, а также динамика формирования общих и профессиональных компетенций.

Таблица 2

Компетенции	Результаты обучения	Основные показатели оценки результата	Формы и методы контроля и оценки
	Знает:		
ОК 01, ОК 02, ПК 1.1	3.1. Этапы решения задач на компьютере, свойства и виды алгоритмов, способы их записи.	– воспроизведение основных свойств алгоритмов (дискретность, детерминированность, конечность и др.); – перечисление способов записи алгоритмов (словесный, блок-схема, псевдокод); – характеристика этапов решения задачи на компьютере (постановка, формализация, алгоритмизация, программирование, отладка).	Устный опрос, тестирование, экзамен
ОК 02, ПК 1.1	3.2. Базовые конструкции языков программирования (переменные, типы данных, операторы, циклы, массивы).	– классификация типов данных (простые, структурированные); – описание синтаксиса основных операторов (ветвления, циклы); – объяснение принципов работы с массивами (индексация, размерность).	Тестирование, контрольная работа, экзамен
ОК 01, ПК 2.2	3.3. Принципы структурного, модульного и объектно-ориентированного программирования.	– характеристика принципов модульности (инкапсуляция, разделение интерфейса и реализации); – описание базовых понятий ООП (класс, объект, наследование, полиморфизм); – объяснение преимуществ модульного и ООП-подходов.	Устный опрос, практические занятия, экзамен
ОК 02, ПК 1.1	3.4. Основные принципы визуально-событийного	– перечисление этапов разработки приложений (проектирование,	Защита проекта, экзамен

	управления, этапы разработки приложений.	кодирование, тестирование, внедрение); – описание принципа событийно-управляемого программирования; – характеристика компонентов для создания графического интерфейса пользователя.	
ПК 2.4	3.5. Методы тестирования и отладки программного обеспечения.	– классификация видов тестирования (модульное, интеграционное, системное); – описание методов отладки (точки останова, пошаговое выполнение, анализ дампов); – характеристика инструментов для автоматизации тестирования.	Устный опрос, экзамен
	Умеет:		
ОК 01, ОК 02, ПК 1.1	У.1. Разрабатывать алгоритмы решения задач и реализовывать их на языке программирования.	– построение блок-схем алгоритмов для типовых задач; – написание программного кода на языке высокого уровня в соответствии с разработанным алгоритмом; – выбор оптимальной структуры данных и алгоритма для решения поставленной задачи.	Практические работы, экзаменационные задания
ОК 02, ПК 2.4	У.2. Использовать языки программирования, типы данных, операторы и функции для создания программ.	– корректное объявление переменных различных типов; – использование условных операторов и циклов для организации вычислительного процесса; – применение стандартных функций и библиотек.	Практические работы, тестирование, экзаменационные задания
ОК 01, ОК 02, ПК 2.2	У.3. Применять принципы модульного и объектно-ориентированного	– разбиение программы на подпрограммы (функции/процедуры); – создание классов и объектов, реализация	Практические работы, индивидуальные задания, экзаменационные

	программирования.	наследования и полиморфизма; – использование механизмов инкапсуляции (свойства, модификаторы доступа).	задания
ПК 2.4	У.4. Выполнять отладку и тестирование программного кода.	– использование интегрированной среды разработки (IDE) для пошаговой отладки; – применение точек останова и просмотр значений переменных; – составление тестовых сценариев и наборов тестовых данных для проверки работоспособности программы.	Практические работы, наблюдение, экзаменационные задания
ПК 2.2	У.5. Работать с файловой системой, осуществлять ввод/вывод данных.	– открытие, чтение и запись данных в текстовые и бинарные файлы; – обработка исключений при работе с файловой системой; – выполнение операций с каталогами (создание, удаление, перемещение).	Практические работы, отчеты, экзаменационные задания

2.1. Оценка освоения учебной дисциплины

2.1.1. Формы и методы оценивания

Предметом оценки служат умения и знания, предусмотренные ФГОС по дисциплине **Основы алгоритмизации и программирования**, направленные на формирование общих и профессиональных компетенций.

Таблица 3

Перечень объектов контроля и оценки

Результаты обучения	Основные показатели оценки результата	Оценка (да/нет)
Знает:		
3.1 - этапы решения задач на компьютере, свойства и виды алгоритмов	– перечисление основных свойств алгоритмов; – описание этапов решения задачи; – различие между видами алгоритмов (линейный, разветвляющийся, циклический).	Да/нет
3.2 - базовые конструкции языков программирования	– определение типов данных; – объяснение работы операторов ветвления и циклов; – описание способов объявления и инициализации массивов.	Да/нет
3.3 - принципы модульного и	– объяснение принципов	Да/нет

объектно-ориентированного программирования	инкапсуляции, наследования и полиморфизма; – описание назначения подпрограмм; – понимание различий между классом и объектом.	
3.4 - принципы визуально-событийного управления	– перечисление этапов разработки приложений; – описание механизма событий; – характеристика компонентов пользовательского интерфейса.	Да/нет
3.5 - методы тестирования и отладки ПО	– классификация методов тестирования; – описание приемов отладки программ; – понимание назначения инструментов отладки в IDE.	Да/нет
Умеет:		
У.1 - разрабатывать алгоритмы и реализовывать их	– построение корректной блок-схемы; – написание работоспособного программного кода; – выбор эффективного алгоритма.	Да/нет
У.2 - использовать языки программирования	– объявление переменных нужного типа; – использование операторов для организации вычислений; – применение стандартных функций.	Да/нет
У.3 - применять принципы модульного и ООП	– разбиение программы на функции; – создание классов с конструкторами и методами; – реализация наследования.	Да/нет
У.4 - выполнять отладку и тестирование	– использование отладчика для поиска ошибок; – составление тестовых примеров; – анализ результатов выполнения программы.	Да/нет
У.5 - работать с файловой системой	– корректное чтение и запись файлов; – обработка ошибок ввода/вывода; – выполнение операций с каталогами.	Да/нет

Критерии оценивания:

Оценка компетенции производится по интегральной оценке ОПОР. Каждый ОПОР оценивается 1 или 0, сумма этих оценок дает оценку компетенции: «да» или «нет». Уровень оценки компетенций производится суммированием количества ответов «да» в процентном соотношении от общего количества ответов.

Для перевода баллов в оценку применяется универсальная шкала оценки образовательных достижений.

Таблица 4

Универсальная шкала оценки образовательных достижений

Процент результативности	Оценка уровня подготовки	
	оценка компетенций обучающихся	оценка уровня освоения дисциплин
90 ÷ 100	высокий	отлично
70 ÷ 89	продвинутый	хорошо
50 ÷ 69	пороговый	удовлетворительно
менее 50	не освоены	неудовлетворительно

2.1.2. Матрица оценок образовательных достижений обучающихся**Оценка достижений обучающихся по результатам экзамена**

Группа _____

	Компетенции ОК 01, ОК 02, ОК 03, ОК 04, ОК 05, ОК 09, ПК 1.1, ПК 2.2, ПК 2.4										та х бал л	% выполнен ия	Оценка компетенц ии
Умения и знания	У 1	У 2	У 3	У 4	У 5	З 1	З 2	З 3	З 4	З 5			
Величина баллов	5	5	5	5	5	5	5	5	5	5	50	100 %	отлично
Ф.И.О. обучающего ся													

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ

Для оценивания компетенций:

ОК 01, ОК 02, ОК 03, ОК 04, ОК 05, ОК 09, ПК 1.1, ПК 2.2, ПК 2.4

3.1. Типовые задания для текущего контроля**Тестовый контроль****Инструкция:** Выберите один или несколько правильных вариантов ответа.**Вариант 1**

- Что такое алгоритм?**
 - программа на языке программирования
 - конечная последовательность команд, приводящая от исходных данных к результату
 - компьютерное устройство для вычислений
 - математическая формула
- Какое свойство алгоритма означает, что процесс решения задачи должен быть разбит на последовательность отдельных шагов?**
 - массовость
 - детерминированность
 - дискретность
 - конечность
- Какой оператор используется для организации ветвления в программе?**
 - for
 - while
 - if
 - switch (в некоторых языках)

Вариант 2

1. **Какая структура данных представляет собой упорядоченную совокупность элементов одного типа, имеющих общее имя и доступных по индексу?**
 - а) запись (структура)
 - б) файл
 - в) массив
 - г) переменная
2. **Что такое инкапсуляция в ООП?**
 - а) возможность создания иерархии классов
 - б) объединение данных и методов их обработки внутри класса и скрывание деталей реализации
 - в) способность объекта принимать множество форм
 - г) создание экземпляра класса
3. **Какой цикл гарантированно выполнится хотя бы один раз?**
 - а) for
 - б) while (с предусловием)
 - в) do...while (с постусловием)
 - г) foreach

Критерии оценивания:

$$K = A/P$$

где К – коэффициент усвоения, А – число правильных ответов, Р – общее число вопросов в тесте.

Оценка	Коэффициент К
5 (отлично)	0,91 – 1,00
4 (хорошо)	0,76 – 0,90
3 (удовлетворительно)	0,61 – 0,75
2 (неудовлетворительно)	0,00 – 0,60

Примерные задания для контрольной работы (по разделу 1)

1. Составьте блок-схему и напишите программу на языке (C#/Python) для вычисления значения функции:
 - ~ Если $x < 0$, то $y = |x|$;
 - ~ Если $0 \leq x < 5$, то $y = x^2$;
 - ~ Если $x \geq 5$, то $y = 2x$.
2. Дан одномерный массив из 15 целых чисел, заполненный случайным образом в диапазоне $[-20, 35]$. Найти:
 - ~ Сумму положительных элементов;
 - ~ Количество элементов, равных нулю;
 - ~ Максимальный элемент массива.

Критерии оценивания:

Контрольная работа оценивается по пятибалльной системе:

«Отлично» – работа выполнена полностью, алгоритм верен, программа компилируется и работает без ошибок на всех тестах, оформление кода соответствует требованиям.

«Хорошо» – работа выполнена полностью, но допущены незначительные ошибки (синтаксические или логические), исправленные после указания преподавателя; либо программа работает правильно, но не оптимально.

«Удовлетворительно» – работа выполнена не менее чем на 70%, но содержит существенные ошибки в логике, программа работает неверно на части тестовых данных.

«Неудовлетворительно» – работа выполнена менее чем на 50%, программа не компилируется или выдает неверные результаты на всех тестах.

Примерные темы рефератов

1. История развития языков программирования.
2. Сравнительный анализ компиляторов и интерпретаторов.
3. Основные принципы объектно-ориентированного программирования: инкапсуляция, наследование, полиморфизм.
4. Методы и алгоритмы сортировки данных: сравнительный анализ.
5. Рекурсия: принципы, примеры использования, достоинства и недостатки.
6. Жизненный цикл программного обеспечения.
7. Методологии тестирования программного обеспечения.
8. Системы контроля версий (Git): основы работы и роль в командной разработке.
9. Обзор современных интегрированных сред разработки (IDE).
10. Анализ сложности алгоритмов (O-большое).

Критерии оценивания реферата:

«Отлично» – выполнены все требования к написанию и защите реферата: обозначена проблема и обоснована её актуальность, сделан краткий анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объём, соблюдены требования к внешнему оформлению, даны правильные ответы на дополнительные вопросы.

«Хорошо» – основные требования к реферату и его защите выполнены, но при этом допущены недочёты: имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; имеются упущения в оформлении; на дополнительные вопросы даны неполные ответы.

«Удовлетворительно» – имеются существенные отступления от требований: тема освещена лишь частично; допущены фактические ошибки в содержании; во время защиты отсутствует вывод.

«Неудовлетворительно» – тема реферата не раскрыта, обнаруживается существенное непонимание проблемы.

Типовые задания для практической работы

Практическое занятие №4. Одномерные массивы.

Задание: Напишите программу, которая формирует массив из 10 целых чисел, введенных пользователем с клавиатуры. Программа должна вывести на экран:

1. Исходный массив.
2. Сумму всех элементов массива.
3. Среднее арифметическое элементов массива.
4. Минимальный элемент массива и его индекс.

Практическое занятие №15. Работа с классами. Создание конструкторов.

Задание: Опишите класс «Book» (Книга) с полями: название (string), автор (string), год издания (int), цена (double). Реализуйте:

1. Конструктор по умолчанию.
2. Конструктор с параметрами для инициализации всех полей.
3. Метод Print(), который выводит информацию о книге в консоль.
В методе Main() создайте не менее 3 объектов класса Book (с использованием разных конструкторов) и выведите информацию о них на экран.

Критерии оценивания практической работы:

«Отлично» – работа выполнена в полном объеме, оформление соответствует требованиям, программа работает корректно на всех тестовых данных, студент уверенно защищает работу и отвечает на все вопросы.

«Хорошо» – работа выполнена полностью, но в оформлении есть незначительные недочеты, либо программа содержит мелкие ошибки, исправленные во время защиты.

«Удовлетворительно» – работа выполнена не менее чем на 70%, программа работает с ошибками, студент затрудняется с объяснением алгоритма.

«Неудовлетворительно» – работа не выполнена или выполнена менее чем на 50%, студент не может продемонстрировать работающую программу и объяснить код.

Типовые задания для самостоятельной работы

1. **Задание на проработку конспекта:** Изучить теоретический материал по теме «Алгоритмы сортировки». Составить конспект, в котором привести описание алгоритмов «пузырьковой» сортировки и сортировки вставками, а также примеры их реализации на псевдокоде.
2. **Задание для самоконтроля:** Напишите рекурсивную функцию для вычисления числа Фибоначчи с заданным номером n . (n вводится пользователем).

Критерии оценивания самостоятельной работы:

Оценка «зачтено» выставляется, если задание выполнено полностью, оформлено аккуратно, студент демонстрирует понимание материала при проверке.

Оценка «не зачтено» выставляется, если задание не выполнено или выполнено формально, студент не ориентируется в представленном материале.

3.2. Типовые задания для промежуточной аттестации

(согласно учебному плану – экзамен)

Примерный перечень экзаменационных вопросов

1. Понятие алгоритма. Свойства алгоритмов. Способы записи алгоритмов.
2. Этапы решения задач на компьютере.
3. Классификация языков программирования. Компиляторы и интерпретаторы.
4. Типы данных: простые и структурированные. Объявление переменных.
5. Операторы ветвления (if, switch). Полная и неполная формы.
6. Операторы цикла (for, while, do...while). Их отличия и применение.
7. Одномерные массивы: объявление, инициализация, ввод/вывод, обработка.
8. Двумерные массивы: объявление, инициализация, обработка.
9. Алгоритмы сортировки массивов (пузырьковая, вставками).
10. Понятие подпрограммы (функции/процедуры). Локальные и глобальные переменные.
11. Рекурсия. Примеры рекурсивных алгоритмов.
12. Основные понятия ООП: класс, объект, интерфейс.
13. Принципы ООП: инкапсуляция.
14. Принципы ООП: наследование.
15. Принципы ООП: полиморфизм.
16. Работа со строками. Основные операции.
17. Понятие коллекций. Отличие от массивов.
18. Работа с файлами. Виды файлов. Чтение и запись данных.
19. Основные принципы визуально-событийного программирования.
20. Этапы разработки приложений с графическим интерфейсом.
21. Методы тестирования программного обеспечения.
22. Методы отладки программного кода. Инструменты отладки в IDE.

Примерные экзаменационные задания (практическая часть)

Задание 1 (уровень У.1, У.2). Разработать алгоритм (блок-схему) и написать программу, которая вводит с клавиатуры три числа и определяет, может ли существовать треугольник с такими длинами сторон. (Треугольник существует, если сумма любых двух сторон больше третьей).

Задание 2 (уровень У.3). Создать класс «Студент» с полями: фамилия, имя, номер группы, массив оценок (5 оценок). Реализовать метод для вычисления среднего балла студента. В основной программе создать массив из 3 студентов и вывести список студентов со средним баллом выше 4.0.

Критерии оценивания для экзамена:

«Отлично» – заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять практические задания, предусмотренные программой. В ответе на теоретические вопросы показано знание основной и дополнительной литературы, взаимосвязь основных понятий дисциплины. Практическое задание выполнено в полном объеме, программный код работает без ошибок, студент уверенно отвечает на вопросы по коду и алгоритму решения.

«Хорошо» – заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе практические задания. Теоретический ответ содержит незначительные неточности или пробелы. Практическое задание выполнено, но допущены незначительные ошибки, исправленные после указания преподавателя, либо программа работает неоптимально. Студент ориентируется в коде, но может затрудняться с ответами на уточняющие вопросы.

«Удовлетворительно» – заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы, справляющийся с выполнением практических заданий, но с помощью преподавателя. Теоретический ответ неполный, содержит ошибки. Практическое задание выполнено частично (более 50%) или содержит существенные ошибки, из-за которых программа работает неверно. Студент с трудом объясняет логику своей программы.

«Неудовлетворительно» – выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой практических заданий. Теоретический ответ отсутствует или не раскрывает вопроса. Практическое задание не выполнено или выполнено менее чем на 50%, программа не компилируется. Студент не может продолжить обучение без дополнительных занятий по дисциплине.

Образец оформления титульного листа для комплекта КИМ (текущий контроль)
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Арктический государственный агротехнологический университет»
Колледж технологий и управления
Цикловая комиссия информационных технологий и программирования

**Комплект контрольно-измерительных материалов
для текущего контроля**
по дисциплине **Основы алгоритмизации и программирования**
по специальности **09.02.11 Разработка и управление программным обеспечением**

Образец оформления титульного листа для комплекта КОС (промежуточная аттестация)

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования

«Арктический государственный агротехнологический университет»

Колледж технологий и управления

Цикловая комиссия информационных технологий и программирования

**Комплект контрольно-оценочных средств
для промежуточной аттестации по результатам освоения дисциплины
по дисциплине Основы алгоритмизации и программирования
по специальности 09.02.11 Разработка и управление программным обеспечением**