

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 25

УТВЕРЖДАЮ

Руководитель образовательной программы

доц., к.т.н., доц.
(должность, уч. степень, звание)

Н.В. Марковская
(инициалы, фамилия)

(подпись)
«27» февраля 2025 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Основы программирования»
(Наименование дисциплины)

Код направления подготовки/ специальности	11.03.02
Наименование направления подготовки/ специальности	Инфокоммуникационные технологии и системы связи
Наименование направленности	Коммуникационные технологии Интернета вещей
Форма обучения	очная
Год приема	2025

Лист согласования рабочей программы дисциплины

Программу составил (а)

доц., к.т.н. 26.02.2025 Е.М. Линский
(должность, уч. степень, звание) (подпись, дата) (инициалы, фамилия)

Программа одобрена на заседании кафедры № 25

«26» февраля 2025 г, протокол № 7/2024-25

Заведующий кафедрой № 25

д.т.н., проф. 26.02.2025 А.М. Тюрликов
(уч. степень, звание) (подпись, дата) (инициалы, фамилия)

Заместитель директора института №2 по методической работе

доц., к.т.н., доц. 26.02.2025 Н.В. Марковская
(должность, уч. степень, звание) (подпись, дата) (инициалы, фамилия)

Аннотация

Дисциплина «Основы программирования» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/специальности 11.03.02 «Инфокоммуникационные технологии и системы связи» направленности «Коммуникационные технологии Интернета вещей». Дисциплина реализуется кафедрой «№25».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ОПК-3 «Способен применять методы поиска, хранения, обработки, анализа и представления в требуемом формате информации из различных источников и баз данных, соблюдая при этом основные требования информационной безопасности»

ОПК-4 «Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности»

ОПК-5 «Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения»

Содержание дисциплины охватывает круг вопросов, связанных с классическими алгоритмами, анализом их сложности и реализации их на языке C++.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, курсовое проектирование, консультации.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена.

Общая трудоемкость освоения дисциплины составляет 9 зачетных единиц, 324 часа.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью курса является изучение классических компьютерных алгоритмов и их реализация на языке C++. В алгоритмической части курса основное внимание уделяется следующим разделам: сортировка, алгоритмы на графах, перебор и методы его сокращения. В части курса, посвященной реализации алгоритмов на C++, акцент сделан на объектно-ориентированном и обобщенном программировании, а также механизме рекурсии.

1.2. Дисциплина входит в состав обязательной части образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Общепрофессиональные компетенции	ОПК-3 Способен применять методы поиска, хранения, обработки, анализа и представления в требуемом формате информации из различных источников и баз данных, соблюдая при этом основные требования информационной безопасности	ОПК-3.У.1 умеет решать задачи обработки данных с помощью средств вычислительной техники
Общепрофессиональные компетенции	ОПК-4 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ОПК-4.3.1 знает современные интерактивные программные комплексы и основные приемы обработки экспериментальных данных, в том числе с использованием стандартного программного обеспечения, пакетов программ общего и специального назначения ОПК-4.У.1 умеет использовать возможности вычислительной техники и программного обеспечения для решения задач управления и алгоритмизации процессов обработки информации ОПК-4.В.1 владеет методами компьютерного моделирования физических процессов при передаче информации, техникой инженерной и компьютерной графики
Общепрофессиональные	ОПК-5 Способен	ОПК-5.3.1 знает методы алгоритмизации,

компетенции	разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	языки и технологии программирования, пригодные для практического применения ОПК-5.У.1 умеет применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач ОПК-5.В.1 владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач
-------------	--	--

2. Место дисциплины в структуре ОП

Дисциплина базируется на знаниях, ранее приобретенных студентами при изучении следующих дисциплин:

- Информатика

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и используются при изучении других дисциплин:

- Моделирование инфокоммуникационных систем
- Основы построения инфокоммуникационных систем и сетей
- Цифровая обработка сигналов

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам	
		№2	№3
1	2	3	4
Общая трудоемкость дисциплины, ЗЕ/ (час)	9/ 324	5/ 180	4/ 144
Из них часов практической подготовки			
Аудиторные занятия, всего час.	187	102	85
в том числе:			
лекции (Л), (час)	68	34	34
практические/семинарские занятия (ПЗ), (час)	34	34	
лабораторные работы (ЛР), (час)	68	34	34
курсовой проект (работа) (КП, КР), (час)	17		17
экзамен, (час)	90	54	36
Самостоятельная работа, всего (час)	47	24	23
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**)	Экз., Экз.	Экз.	Экз.

Примечание: ** кандидатский экзамен

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.
Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП (час)	СРС (час)
Семестр 2					
Раздел 1. Сортировки и рекурсия	10	8	4		4
Раздел 2. Объектно-ориентированное программирование	8	2	10		4
Текущий контроль	1				5
Раздел 3. Перебор	3	8	4		3
Раздел 4. Шаблоны и контейнеры	9	8	12		4
Раздел 5. Обработка ошибок	3	8	4		4
Итого в семестре:	34	34	34		24
Семестр 3					
Раздел 6. Библиотека алгоритмов C++	4		5		3
Раздел 7. Простые алгоритмы на графах	8		9		2
Текущий контроль	1				3
Раздел 8. Динамическое программирование	8		8		3
Раздел 9. Алгоритмы на графах	8		8		2
Раздел 10. Умные указатели	5		4		3
Выполнение курсовой работы				17	7
Итого в семестре:	34		34	17	23
Итого	68	34	68	17	47

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
1	Раздел 1. Сортировки и рекурсия Тема 1.1. Простые сортировки - указатели на функции - сортировка вставками - сортировка выбором - сортировка пузырьком - бинарный поиск Тема 1.2. Сортировки $n \log n$ - рекурсия - быстрая сортировка - сортировка слиянием - пирамидальная сортировка Тема 1.3. Рекуррентные уравнения - подсчет сложности рекуррентных уравнений общего вида

2	<p>Раздел 2. ООП</p> <p>Тема 2.1 Вектор, список</p> <ul style="list-style-type: none"> - инкапсуляция (public/private) - конструктор, деструктор - перегрузка <p>Тема 2.2. Работа с кучей на C++</p> <ul style="list-style-type: none"> - new/delete - создание объектов в куче - конструктор копий - оператор присваивания <p>Тема 2.3. Наследование и полиморфизм.</p> <ul style="list-style-type: none"> - итераторы - protected - virtual (overriding) - таблица виртуальных функций <p>статическое/динамическое связывание</p>
3	<p>Раздел 3. Перебор</p> <ul style="list-style-type: none"> - реализация перебора - метод ветвей и границ - задача о ферзях задача о рюкзаке
4	<p>Раздел 4. Шаблоны и контейнеры</p> <p>Тема 4.1. Шаблоны</p> <ul style="list-style-type: none"> - решение в стиле C - шаблонные классы - шаблонные функции - специализация шаблонов - вектор на шаблонах <p>Тема 4.2. Последовательные контейнеры</p> <ul style="list-style-type: none"> - string, vector, list, deque - внутреннее устройство и основные операции - итераторы и их инвалидация <p>Тема 4.3. Ассоциативные контейнеры</p> <ul style="list-style-type: none"> - set, multiset, map, multimap - unordered_set, unordered_map - внутреннее устройство и основные операции итераторы и их инвалидация
5	<p>Раздел 5. Обработка ошибок</p> <ul style="list-style-type: none"> - обработка ошибок в стиле C <p>исключения</p>
6	<p>Раздел 6. Библиотека алгоритмов C++</p> <ul style="list-style-type: none"> - функторы <p>обзор алгоритмов с примерами</p>
7	<p>Раздел 7. Простые алгоритмы на графах</p> <p>Тема 7.1 Обходы</p> <ul style="list-style-type: none"> - стек, очередь - обход в глубину и ширину <p>Тема 7.2. Поиск точек сочленения</p>
8	<p>Раздел 8. Динамическое программирование</p> <ul style="list-style-type: none"> - задача о рюкзаке задача о возведении в степень

9	Раздел 9. Алгоритмы на графах Тема 9.1. Поиск кратчайших путей - алгоритм Беллмана-Форда - алгоритм Дейкстры Тема 9.2. Максимальный поток в транспортной сети
10	10. Умные указатели - scoped_ptr - auto_ptr shared_ptr

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 2					
1	Сортировки и рекурсия	Решение задач	8		1
2	Объектно-ориентированное программирование	Семинар	2		2
3	Перебор	Решение задач	8		3
4	Шаблоны и контейнеры	Семинар	8		4
5	Обработка ошибок	Семинар	8		5
Всего			34		

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 2				
1	Сортировка	4		1
2	Ввод-вывод на C/C++	2		2
3	Работа с динамической памятью	4		2
4	Списки и вектора	4		2
5	Работа с графическими файлами	4		4,5
6	Работа со звуковыми файлами	4		4,5
7	Классы и объекты	4		4,5
8	Перебор	4		3
9	Стандартные контейнеры	4		4
Семестр 3				
1	Полиморфизм	4		6

2	Динамическое программирование. Задача о рюкзаке	4		8
3	Динамическое программирование. Задача о возведении в степень	4		8
4	Шаблоны	4		4
5	Задачи на обходы графы. Обход в ширину	3		7
6	Задачи на обходы графы. Обход в глубину	3		7
7	Умные указатели	4		10
8	Задачи на графы. Поиск кратчайших путей. Алгоритм Беллмана-Форда	4		9
9	Задачи на графы. Поиск кратчайших путей. Алгоритм Дейкстры	4		9
Всего		68		

4.5. Курсовое проектирование/ выполнение курсовой работы

Цель курсовой работы: адаптировать один из изученных алгоритмов к решению конкретной задачи и реализовать его в виде программного комплекса на языке C++

Примерные темы заданий на курсовую работу приведены в разделе 10 РПД.

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 2, час	Семестр 3, час
1	2	3	4
Изучение теоретического материала дисциплины (ТО)	20	14	6
Курсовое проектирование (КП, КР)	7		7
Подготовка к текущему контролю успеваемости (ТКУ)	10	5	5
Подготовка к промежуточной аттестации (ПА)	10	5	5
Всего:	47	24	23

5. Перечень учебно-методического обеспечения

для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. 7-11.

6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров в библиотеке (кроме электронных экземпляров)
004.4 К 84	Крук, Е. А. Методы программирования и прикладные алгоритмы [Текст] : учебное	40

	пособие в 3 ч. Ч. 1 / Е. А. Крук, А. А. Овчинников ; С.- Петерб. гос. ун-т аэрокосм. приборостроения. - СПб. : Изд-во ГУАП, 2014. - 178 с.	
	Дейл, Н. Программирование на C++ [Электронный ресурс] : самоучитель / Н. Дейл, Ч. Уимз, М. Хедингтон. — Электрон. дан. — Москва : ДМК Пресс, 2007. — 672 с. — Режим доступа: https://e.lanbook.com/book/1219 .	
	Алгоритмизация и программирование : Учебное пособие / С.А. Канцедаль. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2013. - 352 с. http://znanium.com/bookread2.php?book=429576	

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
https://e.lanbook.com/	Электронная библиотечная система
https://znanium.com/	Электронная библиотечная система
https://lib.guap.ru/jirbis2/	Библиотека ГУАП

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
1	Microsoft Visual Studio Express

8.2. Перечень информационно-справочных систем,используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
	Не предусмотрено

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Фонд аудиторий ГУАП для проведения занятий лекционного и семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Специализированная мебель; технические средства обучения, служащие для представления учебной информации большой аудитории; переносной набор демонстрационного оборудования	
2	Вычислительная лаборатория Специализированная мебель; технические средства обучения, служащие для представления учебной информации большой аудитории; лабораторное оборудование (ПЭВМ - 12 шт., объединенных в локальную вычислительную сеть с выходом в вычислительную сеть ГУАП и Интернет)	

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Экзамен	Список вопросов к экзамену
Выполнение курсовой работы	Экспертная оценка на основе требований к содержанию курсовой работы по дисциплине.

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 – Критерии оценки уровня сформированности компетенций

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«отлично» «зачтено»	<ul style="list-style-type: none"> – обучающийся глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий.

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«хорошо» «зачтено»	<ul style="list-style-type: none"> – обучающийся твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий.
«удовлетворительно» «зачтено»	<ul style="list-style-type: none"> – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий.
«неудовлетворительно» «не зачтено»	<ul style="list-style-type: none"> – обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
	<ol style="list-style-type: none"> 1. Простые сортировки. Указатели на функции 2. Простые сортировки. Сортировка вставками 3. Простые сортировки. Сортировка выбором 4. Простые сортировки. Сортировка пузырьком 5. Простые сортировки. Бинарный поиск 6. Сортировки $n \log n$. Рекурсия 7. Сортировки $n \log n$. Быстрая сортировка 8. Сортировки $n \log n$. Сортировка слиянием 9. Сортировки $n \log n$. Пирамидальная сортировка 10. Рекуррентные уравнения. Подсчет сложности 11. Рекуррентные уравнения общего вида 12. Вектор, список 13. Работа с кучей на C++ 14. Итераторы 15. Protected 16. Virtual (overriding) 17. Таблица виртуальных функций 18. Статическое/динамическое связывание 19. Решение в стиле C 20. Шаблонные классы 21. Шаблонные функции 	ОПК-3.У.1 ОПК-4.3.1 ОПК-4.У.1 ОПК-4.В.1 ОПК-5.3.1 ОПК-5.У.1 ОПК-5.В.1

	22. Специализация шаблонов 23. Вектор на шаблонах 24. Перебор 25. Реализация перебора 26. Метод ветвей и границ 27. Задача о ферзях 28. Задача о рюкзаке 29. Последовательные контейнеры. 30. Последовательные контейнеры. Внутреннее устройство и основные операции 31. Последовательные контейнеры. Итераторы и их инвалидация 32. Ассоциативные контейнеры. 33. Ассоциативные контейнеры. Внутреннее устройство и основные операции 34. Ассоциативные контейнеры. Итераторы и их инвалидация 35. Обработка ошибок 36. Обработка ошибок в стиле C 37. Исключения 38. Обзор алгоритмов с примерами 39. Простые алгоритмы на графах 40. Стек, очередь 41. Поиск в глубину 42. Поиск в ширину 43. Поиск точек сочленения 44. Динамическое программирование 45. Задача о возведении в степень 46. Поиск кратчайших путей. Алгоритм Беллмана-Форда 47. Поиск кратчайших путей. Алгоритм Дейкстры 48. Максимальный поток в транспортной сети 49. Умные указатели	
--	---	--

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код индикатора
	Учебным планом не предусмотрено	

Перечень тем для курсового проектирования/выполнения курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для курсового проектирования/выполнения курсовой работы

№ п/п	Примерный перечень тем для курсового проектирования/выполнения курсовой работы
	1. Алгоритм Хаффмана 2. Игра в гексагон (Гексагон) 3. Пентамино 4. Задача коммивояжера 5. Крисс-кросс 6. Игра в каллах 7. Коды, сохраняющие разности

8.	Задача о назначениях
9.	Раскраска графа
10.	Умножение с помощью Бпф
11.	Алгоритм Дейкстры за $O(E \log(V))$ операций
12.	Алгоритм Фано
13.	Алгоритм Рабина-Карпа
14.	Выпуклая оболочка
15.	Цикл отрицательной стоимости
16.	Топологическая сортировка
17.	Максимальный поток в транспортной сети
18.	Максимальное паросочетание в двудольном графе
19.	Алгоритм Прима
20.	Алгоритм Крускала
21.	Алгоритм бинарного возведения в степень
22.	Алгоритм поиска точек сочленения
23.	Выход из лабиринта
24.	Архиватор английских букв
25.	Задача коммивояжера перебором
26.	Несбалансированное дерево поиска
27.	Точка внутри многоугольника

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора
	Не предусмотрено	

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
1	<p>Вариант 1</p> <p>1. Какое значение вернет <code>foo(7)</code>? [3 Балла].</p> <pre>int foo(int n) { if (n <= 0) return 1; else return foo(n/2) + foo(n - 2); }</pre> <p>2. Написать программу, которая выписывает все способы разложить число S на слагаемые, меньшие S, без повторов. [7 Баллов]</p> <p>Пример. <code>allSums(5)</code> --- все способы разложить число 5. Вывод на экран может быть не такой красивый.</p> <pre>5 = 4 + 1 5 = 3 + 2 5 = 3 + 1 + 1 5 = 2 + 2 + 1 5 = 2 + 1 + 1 + 1</pre>

	<p>$5 = 1 + 1 + 1 + 1 + 1$ (Например, $1 + 4$ нельзя, это повтор)</p> <p>3. Дано n предметов, каждый предмет имеет порядковый номер. Написать программу, которая выписывает все возможные сочетания k из n предметов ($k < n$). [10 Баллов].</p> <p>-----</p> <p>Вариант 2</p> <p>1. Какое значение вернет <code>foo(7)</code>? [3 Балла]. <code>int foo(int n) { if (n <= 0) return 1; else } return foo(n - 1) + foo(n - 2) + 1;</code></p> <p>2. Написать программу, которая выписывает все способы разложить число S на $\leq N$ слагаемых, меньших S. Повторы допустимы. [7 Баллов] Примеры <code>allSums(5, 3)</code> --- все способы разложить число 5 на ≤ 3 слагаемых. $5 = 4 + 1$ $5 = 3 + 2$ $5 = 3 + 1 + 1$ $5 = 2 + 3$ $5 = 2 + 2 + 1$ $5 = 2 + 1 + 2$ $5 = 1 + 4$ $5 = 1 + 3 + 1$ $5 = 1 + 2 + 2$ $5 = 1 + 1 + 3$ (Например, $5 = 2 + 1 + 1 + 1$ --- не подходит, т.к. 4 слагаемых)</p> <p>3. Дано n предметов, каждый предмет имеет порядковый номер. Написать программу, которая выписывает все возможные размещения k из n предметов ($k < n$). [10 Баллов].</p>
2	<p>В контрольной можно либо решить только числовые задачи (рюкзак и черепашку) и получить половину баллов. Либо решить задачу о черепашке и задачу на программирование и получить полный балл.</p> <p>\problem Есть сетка (рис. \ref{grid}), по которой может двигаться черепашка. Двигаться можно только по черным линиям. Черепашке необходимо попасть из пункта А в пункт В. На каждом углу она может поворачивать только на Т1 или только на Т2. Время движения по каждой улице указано в таблице. Требуется найти максимальное время, за которое Черепашка может попасть из пункта А в пункт В.</p> <p>Произвести расчет на бумаге методом динамического программирования. Процесс вычисления представить в виде таблицы. Для второй сверху строки таблицы выписать подробный расчет.</p> <p>\begin{figure}[h!] \begin{center} \includegraphics[scale=0.5]{turtle.eps} \caption{Сетка}</p>

```

\label{grid}
\end{center}
\end{figure}
{\vskip 2cm}
\textbf{Вариант 1}:\\
А -- правый верхний угол. В -- левый нижний угол.
Т1 -- запад. Т2 -- юг.

\textbf{Вариант 2}:\\
А -- левый верхний угол. В -- правый нижний угол.
Т1 -- восток. Т2 -- юг.

\problem В рюкзак загружаются предметы  $n$  различных типов (количество предметов каждого типа не ограничено). Максимальный вес рюкзака  $W$ . Каждый предмет типа  $i$  имеет вес  $w_i$  и стоимость  $v_i$  ( $i = 1, 2, \dots, n$ ). Требуется определить максимальную стоимость груза, вес которого не превышает  $W = 6$ .

Произвести расчет на бумаге методом динамического программирования. Процесс вычисления представить в виде графа.

\textbf{Вариант 1}:\\
\begin{tabular}{ccc}
i &  $w_i$  &  $v_i$  \\
1 & 2 & 10 \\
2 & 3 & 30 \\
3 & 1 & 70 \\
4 & 4 & 140
\end{tabular}

\textbf{Вариант 2}:\\
\begin{tabular}{ccc}
i &  $w_i$  &  $v_i$  \\
1 & 2 & 15 \\
2 & 3 & 90 \\
3 & 1 & 30 \\
4 & 4 & 100
\end{tabular}

\problem Написать программу, которая использует метод динамического программирования. Даны  $N$  целых чисел  $X_1, X_2, \dots, X_N$  ( $1 \leq N \leq 10000$ ,  $1 \leq X_i \leq 60000$ ). Требуется вычеркнуть из них минимальное количество чисел так, чтобы оставшиеся шли в порядке Р.

\textbf{Вариант 1}:\\
Р -- в порядке возрастания. \textbf{Пример.} \\
Ввод: 2 5 3 4 6 1 \\
Вывод: 2 3 4 6

\textbf{Вариант 2}:\\
Р -- в порядке убывания. \textbf{Пример.} \\
Ввод: 2 5 3 4 6 1

```

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

- получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;
- получение опыта творческой работы совместно с преподавателем;
- развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.
- появление необходимого интереса, необходимого для самостоятельной работы;
- получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;
- научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);
- получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

- Постановка задачи
- Пример задачи
- Интуитивная идея алгоритма
- Пример работы алгоритма
- Псевдокод
- Сложность
- Выбор средств языка C++ для реализации
- Реализация на C++

Для развития у студентов навыков самостоятельного овладения теоретическим материалом ряд тем дисциплины на лекционных занятиях дается обзорно, что предполагает их самостоятельное детальное изучение.

11.2. Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающаяся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающимся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимися практических занятий:

- закрепление, углубление, расширение и детализация знаний при решении конкретных задач;
- развитие познавательных способностей, самостоятельности мышления, творческой активности;
- овладение новыми методами и методиками изучения конкретной учебной дисциплины;
- выработка способности логического осмысления полученных знаний для выполнения заданий;
- обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

Требования к проведению практических занятий

Практические задания с вариативностью исходных данных выдает преподаватель.

Задание включает расчеты, выполняемые с применением соответствующего математического аппарата.

11.3. Методические указания для обучающихся по выполнению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен углубить и закрепить знания, практические навыки, овладеть современной методикой и техникой эксперимента в соответствии с квалификационной характеристикой обучающегося. Выполнение лабораторных работ состоит из экспериментально-практической, расчетно-аналитической частей и контрольных мероприятий.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач обучающегося:

- приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;
- закрепление, развитие и детализация теоретических знаний, полученных на лекциях;
- получение новой информации по изучаемой дисциплине;
- приобретение навыков самостоятельной работы с лабораторным оборудованием и приборами.

Задание и требования к проведению лабораторных работ

Вариант задания по каждой лабораторной работе обучающийся получает в соответствии с номером в списке группы. Перед проведением лабораторной работы обучающемуся следует внимательно ознакомиться с методическими указаниями по ее выполнению. В соответствии с заданием обучающийся должен подготовить необходимые данные, получить от преподавателя допуск к выполнению лабораторной работы, выполнить указанную последовательность действий, получить требуемые результаты, защитить полученные результаты.

Структура и форма отчета о лабораторной работе

Отчет о лабораторной работе должен включать в себя: титульный лист, цель

работы, формулировку задания, алгоритмы программ, тексты программ и выводы по лабораторной работе.

Требования к оформлению отчета о лабораторной работе

По лабораторным работам выполняется отчет. Титульный лист оформляется в соответствии с шаблоном (образцом), приведенным на сайте ГУАП (<https://new.guap.ru/>) в разделе «Нормативная документация» (<https://guap.ru/standart/doc>). Текстовые и графические материалы оформляются в соответствии с действующими ГОСТами и требованиями, приведенными на сайте ГУАП в разделе «Нормативная документация» (<https://guap.ru/standart/doc>).

Методические указания по прохождению лабораторных работ:

Линский Е.М. Методические указания к выполнению лабораторных работ по дисциплине «Основы программирования». Электронный ресурс кафедры №25

11.4. Методические указания для обучающихся по прохождению курсового проектирования/выполнения курсовой работы

Курсовой проект/ работа проводится с целью формирования у обучающихся опыта комплексного решения конкретных задач профессиональной деятельности.

Курсовой проект/ работа позволяет обучающемуся:

- систематизировать и закрепить полученные теоретические знания и практические умения по дисциплине «Основы программирования» в соответствии с требованиями к уровню подготовки, установленными программой учебной дисциплины, программой подготовки бакалавра по направлению 11.03.02 «Инфокоммуникационные технологии и системы связи»;

- применить полученные знания, умения и практический опыт при решении комплексных задач, в соответствии с основными видами профессиональной деятельности по направлению 11.03.02 «Инфокоммуникационные технологии и системы связи»;

- углубить теоретические знания в соответствии с заданной темой;

- сформировать умения применять теоретические знания при решении нестандартных задач;

- приобрести опыт аналитической, расчётной, конструкторской работы и сформировать соответствующие умения;

- сформировать умения работы со специальной литературой, справочной, нормативной и правовой документацией и иными информационными источниками;

- сформировать умения формулировать логически обоснованные выводы, предложения и рекомендации по результатам выполнения работы;

- развить профессиональную письменную и устную речь обучающегося;

- развить системное мышление, творческую инициативу, самостоятельность, организованность и ответственность за принимаемые решения;

- сформировать навыки планомерной регулярной работы над решением поставленных задач.

Структура пояснительной записки курсовой работы

1. Постановка задачи

2. Алгоритм

3. Псевдокод и анализ сложности

4. Инструкция пользователя

5. Примеры

Способы реализации курсовых работ

Все курсовые работы по данной дисциплине связаны с разработкой программного обеспечения. Данные работы реализуются на языке программирования C/C++/C#.

Защита курсовой работы

Защита курсовой работы является заключительным этапом курсового проектирования. Защита курсовой работы является обязательной и проводится за счет времени, предусмотренного на выполнение работы.

Сроки защиты сообщаются студентам заранее, при выдаче задания. Защита должна проводиться не позднее середины последней недели перед началом сессии.

Требования к оформлению пояснительной записки курсовой работы

1. Оформление с использованием стилей
2. MS Word (OO Writer) или TeX
3. Наличие оглавления
4. Наличие ссылок на литературу
5. Наличие подписей к картинкам

Титульный лист оформляется в соответствии с шаблоном (образцом), приведенным на сайте ГУАП (<https://new.guap.ru/>) в разделе «Нормативная документация» (<https://guap.ru/standart/doc>). Текстовые и графические материалы оформляются в соответствии с действующими ГОСТами и требованиями, приведенными на сайте ГУАП в разделе «Нормативная документация» (<https://guap.ru/standart/doc>).

Для выполнения курсовой работы используется электронный ресурс каф.25:

Линский Е.М. Методические указания по курсовой работе по дисциплине «Основы программирования».

11.5. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Для обучающихся по заочной форме обучения, самостоятельная работа может включать в себя контрольную работу.

В процессе выполнения самостоятельной работы, у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет им развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине.

11.6. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Результаты текущего контроля учитываются при проведении промежуточной аттестации в соответствии с требованиями СТО ГУАП. СМК 3.76 «Положение о текущем контроле успеваемости и промежуточной аттестации студентов и аспирантов ГУАП, обучающихся по образовательным программам высшего образования».

11.7. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

– экзамен – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

При явке на экзамен обучающийся обязан иметь при себе зачетную книжку, которую он предъявляет преподавателю. Прием экзамена без зачетной книжки не допускается. Экзамен проводится в устной форме по экзаменационным билетам, составленным из определяемого в п. 10.3 перечня вопросов к экзамену, утвержденным на заседании кафедры, и подписанным преподавателем – экзаменатором и заведующим кафедрой. При проведении экзамена в устной форме экзаменатору предоставляется право задавать обучающимся уточняющие вопросы. По результатам экзамена положительная оценка («отлично», «хорошо», «удовлетворительно») заносится в ведомость и зачетную книжку. Оценка «неудовлетворительно» заносится только в ведомость. Отсутствие обучающегося на экзамене отмечается в экзаменационной ведомости словами «не явился», либо «н/я». Если со стороны обучающегося во время экзамена допущены нарушения учебной дисциплины (списывание, несанкционированное использование средств мобильной связи, аудио–плееров и других технических устройств), нарушения правил внутреннего распорядка ГУАП, предпринята попытка подлога документов, преподаватель вправе удалить обучающегося с экзамена с занесением в ведомость оценки «неудовлетворительно».

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой