


ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ УЧРЕЖДЕНИЕ
«МУРМАНСКИЙ КООПЕРАТИВНЫЙ ТЕХНИКУМ»

УТВЕРЖДАЮ

 Директор техникума

И.Э. Прокопьева

« 31 » авг 20 23 г.

Приказ № 26 от 31.08 2023 г.

КОМПЛЕКТ ФОНДОВ ОЦЕНОЧНЫХ СРЕДСТВ

по учебной дисциплине

ОПЦ.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

профессиональной образовательной программы
специальности СПО:

09.02.07 Информационные системы и программирование

по программе базовой подготовки

Мурманск, 2023

ОДОБРЕНА

на заседании объединенной цикловой
комиссии

«10» 10 20 22 г.

Протокол № 2

Председатель цикловой комиссии

Родина И.В.

ФИО

подпись

СОГЛАСОВАНО

зам. директора по УМР

ПОЧУ «МКТ»

Худик И.А./

ФИО

подпись

« 05 » 05 20 22 г.

Комплект фондов оценочных средств по учебной дисциплине «Основы алгоритмизации и программирования» разработан на основе ФГОС СПО по специальности среднего профессионального образования 09.02.07 Операционные системы и программирование по программе базовой подготовки и рабочей программы учебной дисциплины.

Организации разработчики: Профессиональное образовательное частное учреждение «Мурманский кооперативный техникум»

Разработчики: Тёлушкин Дмитрий Павлович, преподаватель ПОЧУ «МКТ»

Эксперты от работодателей: _____

Ф.И.О., должность, место работы

СОДЕРЖАНИЕ

| | |
|--|----------|
| I Паспорт комплекта фондов оценочных средств | 4 |
| <i>Область применения</i> | |
| <i>Объекты оценивания – результаты освоения УД</i> | |
| <i>Формы контроля и оценки результатов освоения УД</i> | |
| <i>Система оценивания комплекта ФОС текущего контроля и промежуточной аттестации</i> | |
| | |
| II Текущий контроль и оценка результатов обучения УД..... | 8 |
| <i>Специфика практических работ</i> | |
| <i>Тестовые задания по теме (разделу)</i> | |
| | |
| III Промежуточная аттестация по УД..... | 9 |
| <i>Спецификация экзамена</i> | |

1. Паспорт комплекта фондов оценочных средств

1. Область применения

Комплект фондов оценочных средств (ФОС) предназначен для проверки результатов освоения учебной дисциплины (УД) «Основы алгоритмизации и программирования» профессиональной образовательной программы по специальности СПО 09.02.07 Информационные системы и программирование.

Объем часов на аудиторную нагрузку по УД – 136 часов, на самостоятельную работу – 8 часа.

2. Объекты оценивания – результаты освоения УД

Комплект ФОС позволяет оценить следующие результаты освоения учебной дисциплины в соответствии с ФГОС специальности 09.02.07 Информационные системы и программирование и рабочей программой дисциплины «Основы алгоритмизации и программирования»:

умения:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.
- Выполнять проверку, отладку кода программы.

знания:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения

Вышеперечисленные умения, знания и практический опыт направлены на формирование у студентов следующих **профессиональных и общих компетенций**:

| Код | Наименование результата обучения |
|--------|---|
| ПК 1.1 | Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием |
| ПК 1.2 | Разрабатывать программные модули в соответствии с техническим заданием |
| ПК 1.3 | Выполнять отладку программных модулей с использованием специализированных программных средств |
| ПК 1.4 | Выполнять тестирование программных модулей |
| ПК 1.5 | Осуществлять рефакторинг и оптимизацию программного кода |
| ПК 2.4 | Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения |
| ПК 2.5 | Производить инспектирование компонент программного обеспечения |

| | |
|-------|--|
| | на предмет соответствия стандартам кодирования |
| ОК 1 | Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам |
| ОК 2 | Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности |
| ОК 4 | Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами |
| ОК 5 | Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста |
| ОК 9 | Использовать информационные технологии в профессиональной деятельности |
| ОК 10 | Пользоваться профессиональной документацией на государственном и иностранном языках |

3. Формы контроля и оценки результатов освоения УД

Контроль и оценка результатов освоения – это выявление, измерение и оценивание знаний, умений и формирующихся общих и профессиональных компетенций в рамках освоения УД.

В соответствии с учебным планом специальности 09.02.07 Информационные системы и программирование и рабочей программой дисциплины «Основы алгоритмизации и программирования» предусматривается текущий и промежуточный контроль результатов освоения.

3.1 Формы текущего контроля

Текущий контроль успеваемости представляет собой проверку усвоения учебного материала, регулярно осуществляемую на протяжении курса обучения.

Текущий контроль результатов освоения УД в соответствии с рабочей программой происходит при использовании следующих обязательных форм контроля:

- выполнение и защита практических работ,
- проверка выполнения самостоятельной работы студентов.

Во время проведения учебных занятий дополнительно используются следующие формы текущего контроля – устный опрос, решение задач, тестирование по темам отдельных занятий.

Выполнение и защита практических работ.

Практические работы проводятся с целью усвоения и закрепления практических умений и знаний, овладения профессиональными компетенциями. В ходе практической работы студенты приобретают умения, предусмотренные рабочей программой УД, учатся анализировать полученные результаты и делать выводы, опираясь на теоретические знания.

Список практических работ:

Практическая работа №1. Знакомство со средой программирования. Составление программ линейной структуры.

Практическая работа №2. Составление программ разветвляющейся структуры

Практическая работа №3. Составление программ циклической структуры

Практическая работа №4. Обработка одномерных массивов

Практическая работа №5. Обработка двумерных массивов

Практическая работа №6. Работа со строками

Практическая работа №7. Файлы последовательного доступа

Практическая работа №8. Типизированные файлы

- Практическая работа №9.** Организация процедур
Практическая работа №10. Организация функций
Практическая работа №11. Применение рекурсивных функций
Практическая работа №12. Создание библиотеки подпрограмм
Практическая работа №13. Использование указателей для организации связанных списков
Практическая работа №14. Изучение основных свойств компонентов группы Standard.
Практическая работа №15. Построение простейших геометрических фигур
Практическая работа №16. Разработка оконного приложения с несколькими формами.
Практическая работа №17. Работа с таблицами StringGrid, ColorGrid в Delphi.
Практическая работа №18. Создание проекта с использованием компонентов стандартных диалогов и системы меню
Практическая работа №19. Построение движущихся графических объектов (анимация)
Практическая работа №20. Разработка приложения для работы с типизированным файлом
Практическая работа №21. Разработка приложения для работы с текстовым файлом

Содержание, этапы проведения и критерии оценивания практических работ представлены в методических указаниях по проведению практических работ.

Проверка выполнения самостоятельной работы.

Самостоятельная работа направлена на самостоятельное освоение и закрепление студентами практических умений и знаний, овладение профессиональными компетенциями.

Самостоятельная подготовка студентов по УД предполагает следующие виды и формы работы:

- *Систематическая проработка конспектов занятий, учебной и специальной технической литературы.*
- *Самостоятельное изучение материала и конспектирование лекций по учебной и специальной литературе.*
- *Написание и защита доклада; подготовка к сообщению или беседе на занятии по заданной преподавателем теме.*

Задания для выполнения самостоятельной работы, рекомендации по их выполнению и критерии их оценивания представлены в методических рекомендациях по организации и проведению самостоятельной работы студентов.

Сводная таблица по применяемым формам и методам текущего контроля и оценки результатов обучения

| Результаты обучения (освоенные умения, усвоенные знания) | Формы и методы контроля и оценки результатов обучения |
|--|---|
| УМЕНИЯ | |
| <ul style="list-style-type: none"> – Разрабатывать алгоритмы для конкретных задач. – Использовать программы для графического отображения алгоритмов. – Определять сложность работы алгоритмов. – Работать в среде программирования. – Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. – Оформлять код программы в соответствии со стандартом кодирования. – Выполнять проверку, отладку кода программы. | <i>Текущий контроль –устные и письменные опросы, экспертная оценка выполнения ПР</i> |

| | |
|---|---|
| | |
| ЗНАНИЯ | |
| <ul style="list-style-type: none"> – Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; – Эволюцию языков программирования, их классификацию, понятие системы программирования; – Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; – Подпрограммы, составление библиотек подпрограмм; – Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения. | <p><i>Текущий контроль</i> – устные и письменные опросы, тестирование, подготовка реферата, экспертная оценка выполнения самостоятельной работы</p> |
| | <i>Итоговый контроль в форме Экзамена</i> |

3.2 Форма промежуточной аттестации

Промежуточная аттестация по УД «Основы алгоритмизации и программирования» – экзамен, спецификация которого содержится в данном комплекте ФОС.

Студенты, не выполнившие самостоятельные и практические работы, предусмотренные рабочей программой, являются не аттестованными по УД и не освоившими в полном объеме учебную программу.

Студенты, не освоившие в полном объеме учебную программу, не допускаются к промежуточной аттестации.

4. Система оценивания комплекта ФОС текущего контроля и промежуточной аттестации

Система оценивания каждого вида работ описана в соответствующих методических рекомендациях.

При оценивании практической и самостоятельной работы студента учитывается следующее:

- *качество выполнения практической части работы;*
 - *качество оформления отчета по работе;*
 - *качество устных ответов на контрольные вопросы при защите работы.*
- Каждый вид работы оценивается по пяти бальной шкале.

«5» (отлично) – за глубокое и полное овладение содержанием учебного материала, в котором студент свободно и уверенно ориентируется; за умение практически применять теоретические знания, высказывать и обосновывать свои суждения. Оценка «5» (отлично) предполагает грамотное и логичное изложение ответа.

«4» (хорошо) – если студент полно освоил учебный материал, владеет научно-понятийным аппаратом, ориентируется в изученном материале, осознанно применяет теоретические знания на практике, грамотно излагает ответ, но содержание и форма ответа имеют отдельные неточности.

«3» (удовлетворительно) – если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности, в применении теоретических знаний при ответе на практико-ориентированные вопросы; не умеет доказательно обосновать собственные суждения.

«2» (неудовлетворительно) – если студент имеет разрозненные, бессистемные знания, допускает ошибки в определении базовых понятий, искажает их смысл; не может практически применять теоретические знания.

Тест оценивается по пяти бальной шкале следующим образом: стоимость каждого вопроса 1 балл. За правильный ответ студент получает 1 балл. За неверный ответ или его отсутствие баллы не начисляются.

Оценка «5» соответствует 86% – 100% правильных ответов.

Оценка «4» соответствует 73% – 85% правильных ответов.

Оценка «3» соответствует 53% – 72% правильных ответов.

Оценка «2» соответствует 0% – 52% правильных ответов.

II. Текущий контроль и оценка результатов обучения УД по УД «Основы алгоритмизации и программирования»

1. Практические работы

1.1. Назначение практической работы – оценить уровень подготовки студентов по УД «Основы алгоритмизации и программирования» с целью текущей проверки знаний и умений.

1.2. Содержание практической работы определяется в соответствии с рабочей программой УД «Основы алгоритмизации и программирования».

Примерные задания

Практическое задание 1.

В заданной среде программирования изучите назначение основных палеток с инструментами. Определите назначение для объектов свойств вкладок Properties и действий Events. Выполните создание простого проекта, состоящего из формы и сохраните его. Изучите файлы, которые создаются при сохранении проекта.

Практическое задание 2.

Создайте программу в соответствии с индивидуальным заданием имеющую разветвляющийся алгоритм исполнения.

2. Тестовые задания

2.1. Назначение тестовых заданий – оценить уровень подготовки студентов по УД «Основы алгоритмизации и программирования» с целью текущей проверки знаний и умений.

2.2. Содержание тестовых заданий определяется в соответствии с рабочей программой УД «Основы алгоритмизации и программирования»

Примерные тестовые задания

1. Когда необходимо составлять блок-схему программы:

- а) До начала составления самой программы +
- б) В процессе составления программы
- в) После составления программы

2. Наиболее наглядной формой описания алгоритма является структурно-стилизованый метод:

- а) словесное описание алгоритма
- б) представление алгоритма в виде схемы +
- в) язык программирования высокого уровня

3. Перевод программ с языка высокого уровня на язык более низкого уровня обеспечивает программа -:

- а) паскаль
- б) ассемблер
- в) компилятор +

4. В графических схемах алгоритмов стрелки направлений на линиях потоков:

- а) необходимо рисовать, если направление потока снизу вверх и справа налево +
- б) можно рисовать или не рисовать
- в) рисовать не нужно

5. Разработкой алгоритма решения задачи называется:
- а) точное описание данных, условий задачи и ее целого решения
 - б) сведение задачи к математической модели, для которой известен метод решения
 - в) определение последовательности действий, ведущих к получению результатов +
6. Языком высокого уровня является:
- а) Ассемблер
 - б) Фортран +
 - в) Макроассемблер
7. Как называется алгоритм, в котором действия выполняются друг за другом, не повторяясь:
- а) циклическим
 - б) разветвленным
 - в) линейным +
8. Разработке алгоритма предшествует:
- а) постановка задачи, разработка математической модели +
 - б) постановка задачи, разработка математической модели, выбор метода решения
 - в) постановка задачи, выбор метода решения, проектирование программ
9. Символьный тип данных объявляется служебным словом:
- а) STRING
 - б) WORD
 - в) CHAR +
10. В операторе присваивания $summa := \sqrt{x} + 3 * a$ переменными являются:
- а) \sqrt{x}, a
 - б) $a, x, summa$ +
 - в) $summa, \sqrt{x}, a$
11. Процедура INC(x,k):
- а) увеличивает значение переменной x на величину k +
 - б) преобразует десятичное число x в строку из k символов
 - в) уменьшает значение переменной x на величину k
12. Записью действительного числа с плавающей точкой является:
- а) 48.0001
 - б) 1.0E01 +
 - в) -1.0533333
13. Вещественный тип данных объявляется служебным словом:
- а) REAL +
 - б) INTEGER
 - в) LONGINT
14. Оператор цикла с постусловием:
- а) For ... to...do
 - б) While...do
 - в) Repeat... until +
15. Логический тип данных объявляется служебным словом:
- а) BOOLEAN +
 - б) BYTE
 - в) LOGIC

16. Раздел переменных определяется служебным словом:

- а) LABEL
- б) VAR +
- в) TYPE

17. В языке Паскаль пустой оператор помечаться:

- а) может, но в исключительных ситуациях
- б) не может
- в) может +

18. Раздел типов определяется служебным словом:

- а) BEGIN
- б) TYPE +
- в) LABEL

19. Какие из приведенных типов данных относятся к целочисленному типу данных:

- а) comp, double
- б) integer, real
- в) integer, word, longint +

20. Из приведенных операторов описания переменных неправильно объявлены переменные:

- а) var a,b:real;c:real
- б) VAR f,g,d,t:INTEGER;I,t:REAL +
- в) var I,j,max,min: real

III. Промежуточная аттестация по УД «Основы алгоритмизации и программирования»

Экзамена по дисциплине «Основы алгоритмизации и программирования»

Назначение экзамена – оценить уровень подготовки студентов по УД «Основы алгоритмизации и программирования» с целью установления их готовности к дальнейшему усвоению ОПОП специальности 09.02.07 Информационные системы и программирование.

1. Содержание экзамена определяется в соответствии с ФГОС СПО специальности 09.02.07 Информационные системы и программирование, рабочей программой дисциплины «Основы алгоритмизации и программирования»

2. Принципы отбора содержания дифференцированного зачета:

Ориентация на требования к результатам освоения УД «Основы алгоритмизации и программирования», представленным в соответствии с ФГОС СПО специальности 09.02.07 Информационные системы и программирование и рабочей программой УД «Основы алгоритмизации и программирования»:

Общие компетенции:

| Код | Наименование результата обучения |
|------------|---|
| ОК 1 | Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам |
| ОК 2 | Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности |
| ОК 4 | Работать в коллективе и команде, эффективно взаимодействовать с |

| | |
|-------|--|
| | коллегами, руководством, клиентами |
| ОК 5 | Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста |
| ОК 9 | Использовать информационные технологии в профессиональной деятельности |
| ОК 10 | Пользоваться профессиональной документацией на государственном и иностранном языках |

Профессиональные компетенции:

| Код | Наименование результата обучения |
|------------|---|
| ПК 1.1 | Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием |
| ПК 1.2 | Разрабатывать программные модули в соответствии с техническим заданием |
| ПК 1.3 | Выполнять отладку программных модулей с использованием специализированных программных средств |
| ПК 1.4 | Выполнять тестирование программных модулей |
| ПК 1.5 | Осуществлять рефакторинг и оптимизацию программного кода |
| ПК 2.4 | Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения |
| ПК 2.5 | Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования |

уметь:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.
- Выполнять проверку, отладку кода программы.

знать:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения

3. Структура экзамена

3.1 Экзамен состоит из обязательной части, которая содержит 2 вопроса и 1 практическое задание.

3.2 Задания (вопросы) экзамена дифференцируются по уровню сложности. Обязательная часть включает вопросы, составляющие необходимый и достаточный минимум усвоения знаний и умений в соответствии с требованиями ФГОС СПО, рабочей программы УД.

3.3 Задания экзамена предлагаются в традиционной форме.

3.4 Билеты экзамена равноценны по трудности, одинаковы по структуре, параллельны по расположению заданий.

Тематика вопросов обязательной части:

Первый и второй вопросы – теоретические, направленные на проверку знаний.

Третий вопрос – практический, связан с решением практической ситуации.

4. Система оценивания отдельных заданий (вопросов) и экзамена в целом

4.1 Каждый теоретический вопрос экзамена в традиционной форме оценивается по пяти балльной шкале:

«5» (отлично) – за глубокое и полное овладение содержанием учебного материала, в котором студент свободно и уверенно ориентируется; научно-понятийным аппаратом; за умение практически применять теоретические знания, качественно выполнять все виды лабораторных и практических работ, высказывать и обосновывать свои суждения. Оценка «5» (отлично) предполагает грамотное и логичное изложение ответа (в устной или письменной форме) на практико-ориентированные вопросы; обоснование собственного высказывания с точки зрения известных теоретических положений.

«4» (хорошо) – если студент полно освоил учебный материал, владеет научно-понятийным аппаратом, ориентируется в изученном материале, осознанно применяет теоретические знания на практике, грамотно излагает ответ (в устной или письменной форме), но содержание и форма ответа имеют отдельные неточности.

«3» (удовлетворительно) – если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении теоретических знаний при ответе на практико-ориентированные вопросы; не умеет доказательно обосновать собственные суждения.

«2» (неудовлетворительно) – если студент имеет разрозненные, бессистемные знания по междисциплинарным курсам, допускает ошибки в определении базовых понятий, искажает их смысл; не может практически применять теоретические знания.

4.2 Итоговая оценка экзамена определяется как средний балл по всем заданиям (вопросам).

5. Время проведения экзамена

На подготовку к устному ответу на экзамене студенту отводится не более 30 минут. Время устного ответа студента составляет 5-7 минут.

Вопросы для экзамена

Теоретические задания

1. Основные этапы решения задач на ЭВМ. Критерии качества программы.
2. Жизненный цикл программы. Постановка задачи и спецификация программы.
3. Алгоритм и его свойства. Блок-схемы алгоритмов.
4. Типовые алгоритмы организации линейных и разветвленных процессов.
5. Типовые алгоритмы организации циклических вычислительных процессов.

6. Принцип пошаговой детализации при проектировании алгоритмов. Проверка правильности построения алгоритма.
7. Файлы и файловая система. Имена файлов. Каталоги.
8. Общая характеристика языка Си. Возможности и особенности языка Си. Элементы, алфавит языка Си.
9. Программа, функция, оператор. Константы и переменные.
10. Категории типов данных. Представление чисел в ЭВМ с фиксированной и плавающей точкой. Диапазон, точность, и затраты памяти ЭВМ для представления чисел.
11. Представление символов в ЭВМ. Основные типы данных, инициализация данных.
12. Операции и выражения. Выражения и присваивания. Операнды. Операции. Приоритеты операций и порядок вычисления. Побочные эффекты.
13. Преобразование типов. Пустой оператор. Составной оператор.
14. Операторы управления вычислительным процессом. Условные операторы условия if.
15. Операторы циклов (for, while, do... while).
16. Операторы переходов (break, continue, return, goto).
17. Си. Функции. Основные понятия. Определение функции. Вызовы функций.
18. Передача аргументов в функцию и из функции. Определение и описание указателей.
19. Функция main().
20. Файловый ввод-вывод. Ввод и вывод в стандартные потоки. Форматированный ввод и вывод.
21. Си. Обработка файлов. Ввод и вывод одного символа. Ввод и вывод строк.
22. Си. Массивы. Элементарные операции над массивами.
23. Си. Массивы и указатели. Использование массивов при вызове функций.
24. Связь программ с внешним миром. Аргументы командной строки
25. Типы программного обеспечения. Характеристики и области применения разных типов ПО.

Практические задания (примерные)

Практическая ситуация 1.

В среде разработки создайте приложение производящее диалог с пользователем и оперирующее с вводимыми им ответами.

Практическая ситуация 2.

В среде разработки создайте приложение, оперирующее с графическими абстракциями.

Практическая ситуация 3.

В среде разработки создайте приложение «Тестовый редактор», позволяющее пользователю производить базовые операции с текстом.