

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЛГПУ»)**

Структурное подразделение Институт физико-математического
образования, информационных и обслуживающих технологий
Кафедра информационных образовательных технологий и систем

УТВЕРЖДАЮ

Директор ИФМОИОТ

Е.Е. Горбенко

2023 г.



Приложение к рабочей программе учебной дисциплины

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
для проведения текущего контроля и промежуточной аттестации
обучающихся по дисциплине
«Программирование»**

По направлению подготовки 09.03.04 Программная инженерия

Профиль подготовки Программное обеспечение систем и комплексов

Квалификация выпускника – бакалавр

Форма обучения очная, заочная

Курс ОФО – 1 курс, ЗФО – 1 курс

Разработчик

Швыров В.В.

канд. физ.-мат. наук, доцент,
доцент кафедры информационных
образовательных технологий и
систем

Заведующий кафедрой

Д.А. Капустин

Протокол от «24» ноября 2023 г. №8

Луганск, 2023

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

1.1. Область применения

Фонд оценочных средств (ФОС) – неотъемлемая часть рабочей программы дисциплины «Программирование» и предназначен для контроля и оценки образовательных достижений студентов, освоивших программу дисциплины.

1.2. Цели и задачи фонда оценочных средств

Цель ФОС – установить соответствие уровня подготовки обучающегося требованиям ФГОС ВО бакалавриат по направлению подготовки 09.03.04 Программная инженерия, утвержденным приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 920 и Профессиональным стандартом, утвержденным Приказом Министерства труда и социальной защиты Российской Федерации «Об утверждении профессионального стандарта 06.001 «Программист» от 20.07.2022 № 424н.

1.3. Перечень компетенций, формируемых в процессе освоения основной образовательной программы

Процесс освоения дисциплины направлен на формирование следующих компетенций и индикаторов их достижения:

Код по ФГОС ВО	Индикатор достижения
Общепрофессиональные	
ОПК-8. Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием	ОПК-8.1. Знает теоретические основы поиска, хранения, и анализа информации ОПК-8.2. Умеет применять методы поиска и хранения информации с использованием современных информационных технологий ОПК-8.3. Владеет навыками поиска, хранения и анализа информации с использованием современных информационных технологий
Профессиональные	
ПК-1. Способен применять основные методы и инструменты разработки программного обеспечения	ПК-1.1. Знает основные методы и инструменты разработки программного обеспечения ПК-1.2. Умеет использовать основные методы и инструменты разработки программного обеспечения систем и комплексов ПК-1.3. Владеет навыками разработки программного обеспечения компьютерных вычислительных систем

1.4. Этапы формирования компетенций и средства оценивания уровня их сформированности

Этапы формирования компетенций	Компетенции	Контрольно-оценочные средства / способ оценивания
Тема 1. Стадии создания программного продукта	ОПК-8. ПК-1.	Устный опрос

Тема 2. Синтаксис языка C++. Переменные и константы.	ОПК-8. ПК-1.	Устный опрос. Защита лабораторных работ
Тема 3. Типы данных. Преобразование типов	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 4. Условный оператор	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 5. Оператор множественного выбора switch	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 6. Циклы с пред- и постусловием. Цикл for. Операторы break и continue	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 7. Одномерные массивы. Строки	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 8. Двумерные массивы. Массивы строк	ОПК-8. ПК-1.	Устный опрос. Защита лабораторных работ
Тема 9. Функции (Объявление, вызов, аргументы функции)	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 10. Прототипы, перегрузка, шаблоны функций	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 11. Структуры, перечисления, объединения. Массивы структур	ОПК-8. ПК-1.	Устный опрос. Защита лабораторных работ
Тема 12. Указатели. Ссылки.	ОПК-8. ПК-1.	Устный опрос
Тема 13. Динамическая память. Операторы new и delete.	ОПК-8. ПК-1.	Устный опрос
Тема 14. Одномерные и двумерные динамические массивы.	ОПК-8. ПК-1.	Защита лабораторных работ
Тема 15. Работа с файлами.	ОПК-8. ПК-1.	Устный опрос. Защита лабораторных работ
Текущая аттестация	ОПК-8. ПК-1.	Индивидуальное задание
Промежуточная аттестация	ОПК-8. ПК-1.	Экзамен

1.5. Описание показателей формирования компетенций

Код компетенции	Результаты сформированности
ОПК-8.	Знает: теоретические основы поиска, хранения, и анализа информации. Умеет: применять методы поиска и хранения информации с использованием современных информационных технологий. Владеет навыками: поиска, хранения и анализа информации с использованием современных информационных технологий.
ПК-1.	Знает: основные методы и инструменты разработки программного обеспечения. Умеет: использовать основные методы и инструменты разработки программного обеспечения систем и комплексов. Владеет навыками: разработки программного обеспечения компьютерных вычислительных систем.

1.6. Критерии оценивания компетенций на разных этапах их формирования

Вид учебной работы	Количество баллов		
	ОФО	О-ЗФО	ЗФО
Устные ответы на семинарских занятиях			
Выполнение и защита практических / лабораторных	45		45

работ			
Самостоятельная работа	10		10
Иные виды учебной работы (подготовка презентации, написание реферата, решение задач и др.)	10		10
Экзамен	35		35
Всего	100		

Накопительная система оценивания по 100-балльной шкале

Четырехбалльная система оценивания экзамена	100-балльная шкала	Буквенная шкала, соответствующая 100-балльной шкале	Система оценивания зачета
Отлично	90-100	A – отлично – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом сформированы; все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	Зачтено
Хорошо	83-89	B – очень хорошо – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом в основном сформированы; все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному	
Хорошо	75-82	C – хорошо – теоретическое содержание курса освоено полностью; некоторые практические навыки работы с освоенным материалом сформированы недостаточно; все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	
Удовлетворительно	63-74	D – удовлетворительно – теоретическое содержание курса освоено частично, но пробелы не носят существенного характера; необходимые практические навыки работы с освоенным материалом в основном сформированы; большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки	
Удовлетворительно	50-62	E – посредственно – теоретическое содержание курса освоено частично; некоторые практические навыки работы не сформированы, многие предусмотренные учебной программой обучения учебные задания не выполнены либо качество выполненных некоторых из них оценено числом баллов, близким к минимальному	Не зачтено
Неудовлетворительно	21-49	FX – неудовлетворительно – теоретическое содержание курса освоено частично; необходимые практические навыки работы с освоенным материалом не сформированы; большинство предусмотренных учебной программой обучения учебных заданий не выполнено	

		либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительно самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	
Неудовлетворительно	0-20	F – неудовлетворительно – теоретическое содержание курса не освоено; необходимые практические навыки работы не сформированы; все выполненные учебные задания содержат грубые ошибки; дополнительная самостоятельная работа над материалом курса не приведет к какому-либо значимому повышению качества выполнения учебных заданий	

2. КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА

2.1. Оценочные средства текущего контроля

1. Стадии создания программного продукта.
2. Синтаксис языка C++ (Элементы и структура языка).
3. Переменные локальные и глобальные.
4. Константы
5. Библиотеки в C++.
6. Типы данных C++.
7. Преобразование типов
8. Условный оператор if else
9. Оператор множественного выбора switch
10. Циклы с пред- и постусловием while, do while.
11. Цикл for.
12. Операторы break и continue
13. Одномерные массивы. Назначение
14. Двумерные массивы. Назначение
15. Строки. Библиотеки работы со строками.
16. Массивы строк
17. Функции (Объявление, вызов, типы)
18. Аргументы функции
19. Прототипы функций
20. Перегрузка функций
21. Шаблоны функций
22. Структуры. Назначение, синтаксис.
23. Массивы структур.
24. Перечисления
25. Функции для работы с файлами
26. Указатели в C++.
27. Ссылки
28. Указатели и функции
29. Динамическая память.
30. Динамические массивы

Типовые задания по дисциплине

Задание 1. Написать программу.

1.1. Работник зарабатывает X руб. за каждый из 38 часов своей работы. Ему платят в 1.5 раза больше за каждый час сверх 38 часов. Какую сумму он получит, если отработает Y часов?

1.2. Даны 3 числа: X , Y , Z . Найдите произведение среднего и большего из них

1.3. Даны 2 числа a и b . Большее из этих чисел увеличьте в 4 раза, а меньшее – в 2 раза.

Задание 1. Написать программу. Вывести в значения в виде таблицы: $x \mid y$. Значение, t , s запрашиваются у пользователя. Шаг изменения x задать самостоятельно.

№ вариан та	Функция	Интервал изменения аргумента	Количество вычисляемы х значений
-------------------	---------	------------------------------------	--

1	$y = \sqrt[s]{t + x^3}$	0..5	150
2	$y = t \cdot \frac{3 \cdot x^s}{\sqrt{1 + e^{x^3}}}$	-3..3	200
3	$y = \frac{\sin^2(x^t)}{\sqrt{1 + x^3}}$	0..4	250

Задание 1. Написать программу

№ варианта	Задача
1	В матрице размером 10x10 подсчитать сумму элементов в каждой строке
2	В матрице размером 5x5 поменять местами столбцы и строки, результат сохранить в другую матрицу
3	Найти минимальный элемент каждого столбца матрицы 10x10 и определить его адрес

Задание 1. Написать программу

№ варианта	Задача
1	Даны две целочисленные квадратные матрицы порядка 6. Найти последовательность из нулей и единиц b_1, \dots, b_6 такую, что $b_i=1$, когда количество отрицательных и неотрицательных элементов i -й строки первой матрицы совпадает соответственно с количеством отрицательных и неотрицательных элементов i -й строки второй матрицы
2	Дана действительная матрица размера $m \times n$. Определить числа b_1, \dots, b_m , равные суммам элементов строк
3	Дана действительная матрица размера $m \times n$. Определить числа b_1, \dots, b_m , равные значениям средних арифметических элементов строк

Задание 1. Написать функцию согласно заданному варианту, организовать ее вызов в главной функции программы.

№	Задача 1	Переменные
1	$F = \frac{a}{x+2} e^{-bx^2} + \ln(a+bx) + \frac{a+bx}{e^{-bx^2} \operatorname{tg}\left(\frac{x+2}{a}\right)};$	$a = 1,13;$ $b = -0,17;$ $x = 0,92.$
2	$y = e^{-at} \frac{t + \sqrt{t+a}}{t - \sqrt{t-b}} + \frac{\ln \sqrt{t+a}}{\ln \sqrt{t-b}} * tge^{-at};$	$a = 2,7;$ $b = 1,3;$ $t = 2,5.$

3	$z = 2^{-x} \arctg(x + a) - 3^{-bx} \cos(x + b) + \frac{\ln\left(\frac{x+a}{x+b}\right)}{2^{-x} + 3^{-bx}};$	$a = 1,12;$ $b = -0,14;$ $x = 2,49.$
---	--	--

Задание 1. Написать функцию согласно заданному варианту, организовать ее вызов в главной функции программы.

№	Задача 1.
1	Написать функцию для поиска среднего значения элементов массива целых чисел. Параметрами функции является размер массива, возвращаемым значением – среднее значение.
2	Написать функцию, которая меняет местами два элемента массива. Номера меняемых элементов – параметры функции. С ее помощью отсортировать массив методом пузырьков.
3	Написать функцию, которая принимает два числа и возвращает истину, если первое больше второго. Использовать ее для сортировки массива по убыванию методом пузырьков.

Задание 1. Написать программу

1. Дана строка, состоящая из символов латинского алфавита, разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, удалив из него все вхождения первой буквы этого слова (количество пробелов между словами не изменять).
2. Дана строка, состоящая из символов латинского алфавита, разделенных пробелами (одним или несколькими). Определить количество слов, которые начинаются и заканчиваются одной и той же буквой.
3. Дана строка-предложение из символов латинского алфавита. Вывести самое короткое слово в предложении (если таких слов несколько, то вывести первое из них).

Задание 1. Написать программу

1. Дан массив, состоящий из n элементов. Переставить в обратном порядке элементы массива, расположенные между его минимальным и максимальным элементами.
2. Дан массив, состоящий из n элементов. Назовем *серией* группу подряд идущих одинаковых элементов, а *длиной серии* – количество этих элементов (длина серии не может быть равна 1). Заменить каждую серию, длина которой больше k , на один наименьший элемент массива. Если таких серий нет, то массив оставить без изменений.
3. Дан массив, состоящий из n элементов. Назовем *серией* группу подряд идущих одинаковых элементов, а *длиной серии* – количество этих элементов (длина серии не может быть равна 1). Преобразовать массив, увеличив первую серию наибольшей длины на один элемент.

Задание 1. Написать программу

1. Дана матрица размера $n \times m$. Поменять местами столбцы, содержащие минимальный и максимальный элементы матрицы.

2. Дана матрица размера $n \times m$. Поменять местами ее строки так, чтобы их максимальные элементы образовывали возрастающую последовательность.
3. Дана квадратная матрица порядка $2n + 1$. Зеркально отразить ее элементы относительно побочной диагонали матрицы.

2.2. Оценочные средства для промежуточной аттестации

Теоретические вопросы к экзамену

1. Какие стадии создания программного продукта вы знаете?
2. Синтаксис языка C++ (Элементы и структура языка)?
3. Как правильно называть переменные в C++?
4. Какие типы данных существуют?
5. Что такое переменная в C++?
6. Зачем нужны глобальные переменные?
7. Что такое локальная переменная?
8. Что такое область видимости?
9. Что такое константа?
10. Как подключать библиотеки в программе?
11. Отличие глобальных и локальных переменных?
12. Как объявить переменную типа целое число в C++?
13. Как объявить переменную типа строка в C++?
14. Как объявить переменную типа вещественное число в C++?
15. Зачем нужен логический тип?
16. Явное преобразование типов данных
17. Неявное преобразование типов данных.
18. Зачем нужны фигурные скобки в C++?
19. Зачем нужна «;» в C++?
20. Как вводить данные с клавиатуры в C++?
21. Как выводить информацию на экран в C++?
22. Какая разница между операторами "==" и "=" в C++?
23. Зачем нужно «&&» в условном операторе?
24. Зачем нужно «||» в условном операторе?
25. Как в условном операторе можно объединить несколько условий в одно?
26. Что такое условный оператор if в C++ и как он используется?
27. Что такое цикл for в C++ и как он работает?
28. Что такое алгоритм в программировании и какие типы алгоритмы вы знаете?
29. Какие свойства алгоритма вы знаете?
30. Что такое линейный алгоритм и как он выполняется?
31. Что такое условный алгоритм и как он выполняется?
32. Что такое циклический алгоритм и как он выполняется?
33. Что такое блок-схема (flowchart) в программировании?
34. Какие элементы используются в блок-схемах и как они представляются?
35. Какие типы блоков (элементов) могут быть использованы в блок-схемах?
36. Какие символы используются для представления различных типов блоков в блок-схемах?
37. Какие виды соединений между блоками могут быть использованы в блок-схемах?
38. Какие основные правила следует учитывать при создании блок-схем?
39. Какие преимущества предоставляют блок-схемы при разработке программ?
40. Какие основные шаги нужно выполнить для создания блок-схемы?
41. Как читать и интерпретировать блок-схему?
42. Что такое массив в C++ и как его объявить?
43. Что такое указатель в C++ и как его использовать?

44. Как объявить функцию в C++?
45. Что такое прототип функции?
46. Что означает параметр по умолчанию в функции?
47. Зачем нужны константные параметры в функциях?
48. Как передаются параметры в функцию?
49. Каким образом в C++ можно осуществить ввод и вывод данных?
50. Что такое константа в C++ и как её объявить?
51. Что такое перегрузка функций в C++ и как она работает?
52. Какие стандартные библиотеки есть в C++ и для чего они используются?
53. Как объявить и инициализировать одномерный массив в C++?
54. Как объявить и инициализировать двумерный массив в C++?
55. Как перебрать все элементы двумерного массива с помощью одного цикла в C++?
56. Как получить доступ к элементу массива в C++?
57. Как совершить вставку элемента в середину одномерного массива?
58. Как удалить последний элемент в массиве?
59. Как объявить и использовать цикл for в C++?
60. Как объявить и использовать цикл while в C++?
61. Как объявить и использовать цикл do-while в C++?
62. Как определить длину (количество элементов) массива в C++?
63. Как объявить и инициализировать двумерный массив (матрицу) в C++?
64. Как перебрать все элементы двумерного массива с помощью вложенных циклов в C++?
65. Как использовать условный оператор if-else в C++?
66. Как использовать условный оператор switch-case в C++?
67. Как использовать операторы сравнения (>, <, >=, <=, ==, !=) в условных выражениях в C++?
68. Что такое алгоритм в программировании и какие алгоритмы вы знаете?
69. Как отсортировать массив элементов типа int в порядке возрастания с помощью алгоритма сортировки пузырьком в C++?
70. Как отсортировать массив элементов типа int в порядке возрастания с помощью алгоритма сортировки выбором в C++?
71. Как отсортировать массив элементов типа int в порядке возрастания с помощью алгоритма сортировки вставкой в C++?
72. Как отсортировать массив элементов типа string в порядке возрастания с помощью алгоритма сортировки пузырьком в C++?
73. Как отсортировать массив элементов типа string в порядке возрастания с помощью алгоритма сортировки выбором в C++?
74. Как отсортировать массив элементов типа string в порядке возрастания с помощью алгоритма сортировки вставкой в C++?
75. В чем заключается алгоритм быстрой сортировки?
76. Что такое шаблоны функций в C++ и как они используются?
77. Что такое перегрузка функций и как она используется?
78. Как объявить и использовать шаблонную функцию в C++?
79. Какие преимущества использования шаблонных функций в C++?
80. Что такое рекурсия и как она используется в программировании?
81. Как написать рекурсивную функцию для вычисления факториала числа в C++?
82. Как написать рекурсивную функцию для вычисления степени числа в C++?
83. Как написать рекурсивную функцию для вычисления НОК и НОД числа в C++?
84. Что такое передача параметров по значению и по ссылке в функциях C++?
85. Как создать и использовать функцию с переменным числом аргументов (variadic function) в C++?
86. Что такое структура (структура данных) в C++ и как она объявляется?
87. Как объявить переменную типа структуры и получить доступ к ее членам в C++?
88. Что такое перечисление (enum) в C++ и как оно объявляется?

89. Как использовать перечисление в C++?
90. Что такое объединение (union) в C++ и как оно объявляется?
91. Как использовать объединение в C++?
92. Что такое указатель (pointer) в C++ и как он объявляется?
93. Как получить адрес переменной в C++?
94. Как получить значение переменной, на которую указывает указатель, в C++?
95. Что делает оператор изъятия адреса в C++?
96. Что делает оператор разыменовывания в C++?
97. Что такое ссылка (reference) в C++ и как она объявляется?
98. В чем разница между указателями и ссылками в C++?
99. Как объявить и инициализировать строку (массив символов) в C++?
100. Как ввести строку содержащую пробелы с клавиатуры?
101. Как ввести строку содержащую кириллицу с клавиатуры?
102. Как получить длину строки в C++?
103. Как сконкатенировать две строки в C++?
104. Как скопировать одну строку в другую в C++?
105. Как найти заданную подстроку в строке в C++?
106. Какие библиотеки работают со строками?
107. Как проверить, содержит ли строка определенный подстроку в C++?
108. Как найти первое вхождение символа или подстроки в строке в C++?
109. Как осуществить поиск всех вхождений символа или подстроки в строке в C++?
110. Как сравнить две строки?
111. Что такое динамический одномерный массив в C++ и как он создается?
112. Какой оператор необходим для выделения памяти в C++?
113. Какой оператор необходим для освобождения памяти в C++?
114. Как добавить элемент в динамический одномерный массив в C++?
115. Как удалить элемент из динамического одномерного массива в C++?
116. Как изменить размер динамического одномерного массива в C++?
117. Что такое динамический двумерный массив в C++ и как он создается?
118. Как выделить и освободить память при работе с двумерным динамическим массивом?
119. Как добавить элемент в динамический двумерный массив в C++?
120. Как удалить элемент из динамического двумерного массива в C++?
121. Как изменить размер динамического двумерного массива в C++?
122. Какие алгоритмы поиска существуют в C++?
123. Что такое линейный поиск?
124. Что такое бинарный поиск?
125. Найдите заданную пользователем фамилию в массиве фамилий?
126. Посчитайте, сколько раз встречается заданное число пользователем в массиве целых чисел.
127. Каким образом можно работать с файлами в C++?
128. Что такое вектор (vector) в C++ и как он отличается от обычного массива?
129. Как объявить и инициализировать вектор в C++?
130. Как добавить элемент в конец вектора в C++?
131. Как получить доступ к элементам вектора в C++?
132. Как удалить элемент из вектора в C++?
133. Как объявить двумерный вектор?
134. Могут ли быть двумерные массивы с разным количеством элементов в строках?
135. Что такое указатель на функцию в C++?
136. Как объявить указатель на функцию в C++?
137. Что такое лямбда-выражение?
138. Функции работы с файлами: чтение из файла.
139. Функции работы с файлами: запись в файле.

Типовые задания к экзамену

1. Заполнить массив n целыми числами, n – количество, вводит пользователь. Числа задаются случайным образом. Написать функцию, позволяющую добавлять новое число, введенное пользователем, на указанную пользователем позицию. Вывести результат.
2. Заполнить массив n вещественными числами, n – количество, вводит пользователь. Сами числа вводить с клавиатуры. Написать функцию, позволяющую удалять число с заданной пользователем позиции. Вывести результат.
3. Заполнить массив n буквами латинского алфавита, n – количество, вводит пользователь. Написать функцию, позволяющую извлекать гласные буквы в другой массив без повторений. Вывести результат.
4. Заполнить массив n строками, n – количество, вводит пользователь. Написать функцию, позволяющую перевернуть каждую строку задом наперед. Вывести результат.
5. Заполнить массив n целыми числами, числа задаются случайным образом. n – количество, вводит пользователь. Написать функцию, позволяющую заменить все числа меньше среднего арифметического этого массива на 0. Вывести результат.
6. Заполнить массив n вещественными числами, вводятся с клавиатуры. n – количество, вводит пользователь. Написать функцию, позволяющую заменить все числа больше среднего арифметического этого массива на него. Вывести результат.
7. Заполнить массив n целыми числами, числа задаются случайным образом. n – количество, вводит пользователь. Написать функцию, позволяющую заменить все четные числа на их сумму. Отсортировать массив. Вывести результат.
8. Заполнить массив n вещественными числами, вводятся с клавиатуры. n – количество, вводит пользователь. Написать функцию, позволяющую сформировать новый массив, элементы которого это суммы первого и последнего, второго и предпоследнего и т.д. элементов из первого массива. Вывести результат на экран.
9. Заполнить массив n символами. n – количество, вводит пользователь. Написать функцию, позволяющую извлекать символы-буквы латиницы в другой массив. Вывести результат.
10. Заполнить массив n строками. n – количество, вводит пользователь. Написать функцию, позволяющую сформировать из этого массива массив - строки, которого содержат цифры. Вывести результат.
11. Заполнить массив n строками. n – количество, вводит пользователь. Написать функцию, позволяющую сформировать из этого массива массив – строки, которого содержат цифры. Вывести результат.