



Министерство образования Иркутской области
Государственное бюджетное профессиональное
образовательное учреждение Иркутской области
«Иркутский авиационный техникум»

УТВЕРЖДАЮ
Директор
ГБНОУИО «ИАТ»

 Якубовский А.Н.
«08» февраля 2023 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

ОП.06 Основы алгоритмизации и программирования

специальности

09.02.01 Компьютерные системы и комплексы

Иркутск, 2023

Рассмотрена
цикловой комиссией
КС протокол №5 от 07.02.2023
г.

№	Разработчик ФИО
1	Ульянова Екатерина Алексеевна

1. ОБЩИЕ ПОЛОЖЕНИЯ

1.1. Область применения фонда оценочных средств (ФОС)

ФОС по дисциплине является частью программы подготовки специалистов среднего звена по специальности 09.02.01 Компьютерные системы и комплексы

1.2. Место дисциплины в структуре ПССЗ:

ОП.00 Общепрофессиональный цикл.

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

Результаты освоения дисциплины	№ результата	Формируемый результат
Знать	1.1	понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции
	1.2	классификация языков программирования
	1.3	понятие системы программирования
	1.4	основные элементы языка, структура программы
	1.5	методы реализации типовых алгоритмов
	1.6	операторы и операции, управляющие структуры, структуры данных, классы памяти
	1.7	понятие подпрограммы, библиотеки подпрограмм
	1.8	объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения
Уметь	2.1	разрабатывать и анализировать алгоритмы для решения поставленных задач
	2.2	определять сложность алгоритмов
	2.3	реализовывать типовые алгоритмы в виде программ на актуальных языках программирования
	2.4	использовать средства проектирования для создания и графического отображения алгоритмов

	2.5	оформлять код программ в соответствии со стандартом кодирования
	2.6	выполнять проверку, отладку кода программы
Личностные результаты реализации программы воспитания	4.1	Проявляющий и демонстрирующий уважение к труду человека, осознающий ценность собственного труда и труда других людей. Экономически активный, ориентированный на осознанный выбор сферы профессиональной деятельности с учетом личных жизненных планов, потребностей своей семьи, российского общества. Выражающий осознанную готовность к получению профессионального образования, к непрерывному образованию в течение жизни Демонстрирующий позитивное отношение к регулированию трудовых отношений. Ориентированный на самообразование и профессиональную переподготовку в условиях смены технологического уклада и сопутствующих социальных перемен. Стремящийся к формированию в сетевой среде лично и профессионального конструктивного «цифрового следа»
	4.2	Ориентированный на профессиональные достижения, деятельно выражающий познавательные интересы с учетом своих способностей, образовательного и профессионального маршрута, выбранной квалификации
	4.3	Осознающий и деятельно выражающий приоритетную ценность каждой человеческой жизни, уважающий достоинство личности каждого человека, собственную и чужую уникальность, свободу мировоззренческого выбора, самоопределения. Проявляющий бережливое и чуткое отношение к религиозной принадлежности каждого человека, предупредительный в отношении выражения прав и законных интересов других людей
	4.4	Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности

1.4. Формируемые компетенции:

ОК.1 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

ОК.2 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности

ПК.1.1 Анализировать требования технического задания на проектирование цифровых систем

ПК.2.1 Проектировать, разрабатывать и отлаживать программный код модулей управляющих программ

ПК.2.2 Владеть методами командной разработки программных продуктов

2. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

2.1 Текущий контроль (ТК) № 1

Тема занятия: 1.1.4. Составление и оформление блок-схем простых алгоритмов.

Метод и форма контроля: Письменный опрос (Опрос)

Вид контроля: Практическая работа с использованием ИКТ

Дидактическая единица: 1.1 понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции

Занятие(-я):

1.1.1. Понятие алгоритма. Свойства и виды алгоритмов.

1.1.2. Способы описания алгоритмов: псевдокоды. Блок-схема: основные элементы, правила составления. Стандарты графического оформления алгоритмов.

1.1.3. Базовые алгоритмические конструкции: линейная, разветвляющаяся, циклическая. Критерии «хорошего» алгоритма.

Задание №1

Дать развернутые ответы на следующие вопросы:

1. Дайте определение понятию "Алгоритм".
2. Перечислите и объясните свойства алгоритма.
3. Назовите формы записи алгоритмов.
4. Назовите и изобразите 5 основных элементов блок-схем.

<i>Оценка</i>	<i>Показатели оценки</i>
5	На все вопросы даны правильные ответы.
4	На один из вопросов не дан ответ / допущены ошибки в формулировке ответа.
3	На два вопроса не даны ответы / на один вопрос не дан ответ и допущены ошибки в формулировке ответа.

2.2 Текущий контроль (ТК) № 2

Тема занятия: 1.2.6. Проектирование и оформление алгоритмов поиска.

Метод и форма контроля: Практическая работа (Опрос)

Вид контроля: Практическая работа с использованием ИКТ

Дидактическая единица: 1.1 понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции

Занятие(-я):

1.1.4. Составление и оформление блок-схем простых алгоритмов.

1.1.5. Составление и оформление блок-схем простых алгоритмов.

1.2.1. Основные методы и этапы проектирования алгоритмов: постановка задачи, математическое описание – математическая модель. Нисходящее, модульное и восходящее проектирование.

1.2.2.Эффективность и сложность алгоритма, их практическая значимость.

1.2.3.Алгоритмы поиска. Алгоритмы сортировки. Вложенные циклы.

Вспомогательные алгоритмы.

1.2.4.Различные комбинации алгоритмических конструкций. Тестовые данные.

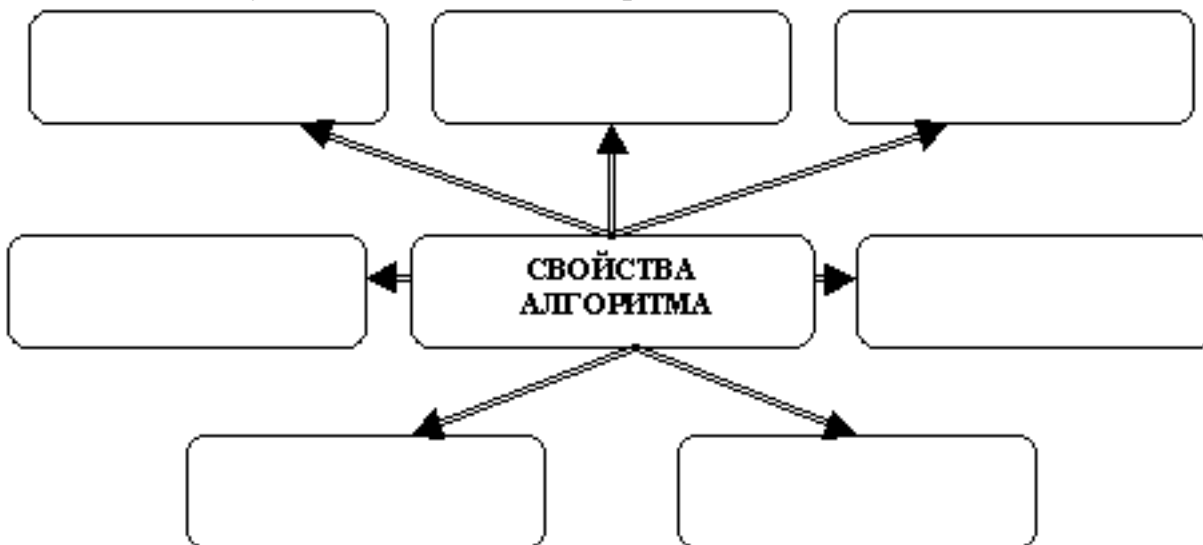
Алгоритм Евклида. Алгоритмы решения нелинейных и линейных уравнений.

Декомпозиция алгоритма.

1.2.5.Проектирование и оформление алгоритмов сортировки.

Задание №1

Заполните схему, вписав свойства алгоритма:



<i>Оценка</i>	<i>Показатели оценки</i>
5	Студент верно записал все свойства алгоритма.
4	Студент записал все свойства алгоритма, кроме одного.
3	Студент записал все свойства алгоритма, кроме трех.

Дидактическая единица: 1.5 методы реализации типовых алгоритмов

Занятие(-я):

1.2.1.Основные методы и этапы проектирования алгоритмов: постановка задачи, математическое описание – математическая модель. Нисходящее, модульное и восходящее проектирование.

1.2.2.Эффективность и сложность алгоритма, их практическая значимость.

1.2.3.Алгоритмы поиска. Алгоритмы сортировки. Вложенные циклы.

Вспомогательные алгоритмы.

1.2.4.Различные комбинации алгоритмических конструкций. Тестовые данные.

Алгоритм Евклида. Алгоритмы решения нелинейных и линейных уравнений.

Декомпозиция алгоритма.

1.2.5.Проектирование и оформление алгоритмов сортировки.

Задание №1

Составить алгоритм следующих задач:

1. Вычислить периметр и площадь прямоугольного треугольника по заданным длинам двух катетов a и b .
2. Найти максимальное значение из трех различных целых чисел, введенных с клавиатуры.
3. Дано двузначное число. Определить: какая из его цифр больше: первая или вторая.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм трех задач составлен верно.
4	Алгоритм двух задач составлен верно.
3	Алгоритм одной задачи составлен верно.

Дидактическая единица: 2.1 разрабатывать и анализировать алгоритмы для решения поставленных задач

Занятие(-я):

- 1.1.4. Составление и оформление блок-схем простых алгоритмов.
- 1.1.5. Составление и оформление блок-схем простых алгоритмов.
- 1.2.5. Проектирование и оформление алгоритмов сортировки.

Задание №1

Построить алгоритм решения задачи в виде блок-схемы.

Задача: Найти среднее арифметическое чисел, введенных с клавиатуры. Признак конца ввода - ноль. Определить количество введенных чисел.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм составлен без ошибок.
4	Алгоритм составлен с одной ошибкой.
3	Алгоритм составлен с двумя ошибками.

2.3 Текущий контроль (ТК) № 3

Тема занятия: 1.2.8. Проектирование и оформление сложных алгоритмов.

Метод и форма контроля: Практическая работа (Опрос)

Вид контроля: Практическая работа с использованием ИКТ

Дидактическая единица: 2.2 определять сложность алгоритмов

Занятие(-я):

- 1.2.5. Проектирование и оформление алгоритмов сортировки.

Задание №1

Построить алгоритм решения задачи в виде блок-схемы:

1. С клавиатуры ввести последовательность символов. Написать функцию,

вычисляющую длину этой последовательности.

2. Сформировать матрицу $\{7,8\}$. Вывести матрицу, найти и вывести значение минимального элемента, вывести координаты всех минимальных элементов матрицы и их количество.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм составлен без ошибок.
4	Алгоритм составлен с одной ошибкой.
3	Алгоритм составлен с двумя ошибками.

2.4 Текущий контроль (ТК) № 4

Тема занятия: 2.2.6.Решение задач на составление циклических алгоритмов.

Метод и форма контроля: Письменный опрос (Опрос)

Вид контроля: Самостоятельная работа

Дидактическая единица: 1.2 классификация языков программирования

Занятие(-я):

2.1.1.Классификация и генеалогия актуальных языков программирования. Понятие системы программирования.

2.1.2.Заполнение таблицы «Классификация языков программирования».

2.1.3.Основные элементы языка. Структура типовой программы. Особенности актуальных сред программирования.

2.1.4.Изучение инструментария среды программирования.

2.1.5.Подготовка структуры программы в среде программирования.

Задание №1

Вопрос 1. Как называется набор правил описания последовательностей символов составляющих программу и какое вычисление производит программа?

- Язык программирования.
- Программа.
- Синтаксис программы.

Вопрос 2. Как называется последовательность символов определяющих вычисления? Данную запись на языках программирования называют исходным кодом.

- Программа.
- Синтаксис программы.

- Язык программирования.

Вопрос 3. Приведите классификация языков программирования в правильном соответствии

Символьные языки, - машинно-ориентированные
максимально приближенные к машинным командам конкретных ПЭВМ. Языки низкого уровня
языки, ни в коей мере не ориентированные на конкретные ПЭВМ. Языки высокого уровня - машинно-независимые языки

<i>Оценка</i>	<i>Показатели оценки</i>
5	Правильно отвечено на 3 вопроса.
4	Правильно отвечено на 2 вопроса.
3	Правильно отвечено на 1 вопрос.

Дидактическая единица: 1.4 основные элементы языка, структура программы
Занятие(-я):

2.1.3.Основные элементы языка. Структура типовой программы. Особенности актуальных сред программирования.

2.1.5.Подготовка структуры программы в среде программирования.

2.2.1.Методы реализации типовых алгоритмов. Переменные: определение, правила именования. Типы данных: значимые и ссылочные. Объявление и инициализация переменных. Область действия и время существования переменных. Константы: определение, виды и правила записи в программе.

2.2.2.Операторы и операции. Понятие выражения. Математические операторы. Старшинство операторов. Математические функции (класс Math). Ввод – вывод данных. Операторы присваивания.

2.2.3.Операторы отношения. Проверка простых и сложных условий. Вложенные условные операторы. Оператор выбора. Операторы перехода.

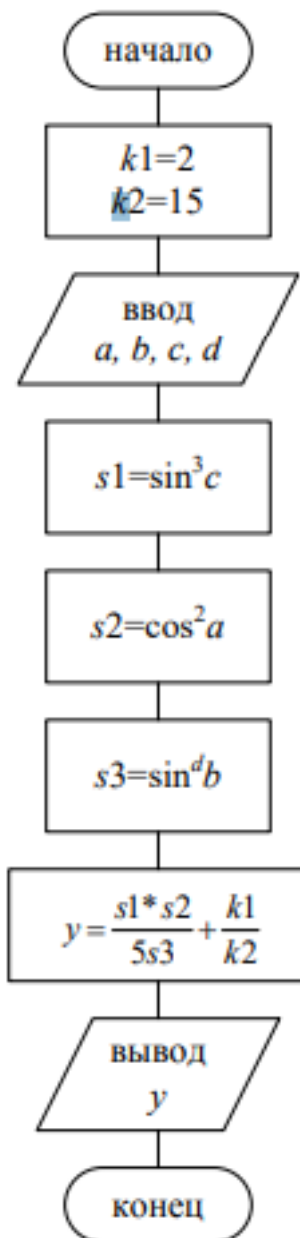
2.2.4.Операторы цикла. Стандартные операции при работе с циклическими алгоритмами. Принудительный выход из цикла.

2.2.5.Реализация простых циклических алгоритмов.

Задание №1

Вычислить значение функции $15 \sin^2 \cos^3 2 + x = b \cdot c \cdot a \cdot y \cdot d$ при $A = 9,5$; $B = 1,365$; $C = 6,5$; $D = 5$. Использовать два варианта ввода исходных данных и вывода

результатов: возможности библиотеки функций языка С и библиотеки классов языка С#. Схема программы для задания представлена на рисунке



<i>Оценка</i>	<i>Показатели оценки</i>
5	Программа полностью соответствует алгоритму.
4	Программа соответствует алгоритму частично.
3	Программа не соответствует алгоритму.

Дидактическая единица: 1.5 методы реализации типовых алгоритмов

Занятие(-я):

1.2.6. Проектирование и оформление алгоритмов поиска.

1.2.7. Проектирование и оформление алгоритмов поиска.

- 1.2.8.Проектирование и оформление сложных алгоритмов.
 1.2.9.Проектирование и оформление сложных алгоритмов.
 1.2.10.Проектирование и оформление сложных алгоритмов.

Задание №1

К задаче необходимо написать программы, используя циклы с предусловием и постусловием.

Задание: Вычислить произведение вводимых пользователем с клавиатуры чисел, если числа считываются до тех пор, пока не будет введена единица.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Студентом написаны две программы, в которых использован цикл с предусловием и постусловием.
4	Студентом написаны две программы, в которых использован цикл с предусловием и постусловием. Допущены незначительные ошибки.
3	Студентом написана одна программа с предусловием/постусловием.

Дидактическая единица: 2.3 реализовывать типовые алгоритмы в виде программ на актуальных языках программирования

Занятие(-я):

2.1.5.Подготовка структуры программы в среде программирования.

2.2.5.Реализация простых циклических алгоритмов.

Задание №1

Написать программу: С помощью генератора случайных чисел сформировать массив D из 18 элементов. Вывести его на экран. Найти сумму положительных элементов массива, произведение отрицательных и количество нулей.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Программа написана верно.
4	Программа написана с одной ошибкой.
3	Программа написана с двумя ошибками.

2.5 Текущий контроль (ТК) № 5

Тема занятия: 3.1.6.Создание классов для обработки массива данных.

Метод и форма контроля: Лабораторная работа (Опрос)

Вид контроля: Самостоятельная работа

Дидактическая единица: 1.6 операторы и операции, управляющие структуры, структуры данных, классы памяти

Занятие(-я):

- 2.2.2.Операторы и операции. Понятие выражения. Математические операторы. Старшинство операторов. Математические функции (класс Math). Ввод – вывод данных. Операторы присваивания.
- 2.2.3.Операторы отношения. Проверка простых и сложных условий. Вложенные условные операторы. Оператор выбора. Операторы перехода.
- 2.2.4.Операторы цикла. Стандартные операции при работе с циклическими алгоритмами. Принудительный выход из цикла.
- 2.2.5.Реализация простых циклических алгоритмов.
- 2.2.6.Решение задач на составление циклических алгоритмов.
- 2.2.7.Решение задач на составление циклических алгоритмов.
- 2.2.8.Массивы: определение, виды. Объявление одномерного массива. Варианты инициализации. Ввод и вывод одномерных массивов. Стандартные операции для работы с массивами. Обработка одномерных и двумерных массивов.
- 2.2.9.Реализация алгоритмов обработки одномерных массивов.
- 2.2.10.Реализация алгоритмов обработки двумерных массивов.
- 2.2.11.Реализация алгоритмов обработки одномерных массивов.
- 2.2.12.Решение задач с использованием функций работы со строками.
- 2.2.13.Управляющие структуры. Понятие потока. Механизм буферизации. Классы памяти. Доступ к файлам.
- 2.2.14.Понятие подпрограммы, библиотеки подпрограмм. Библиотеки среды разработки.
- 2.2.15.Реализация алгоритмов обработки текстовых данных.
- 2.2.16.Реализация сложных алгоритмов поиска и ввода-вывода.
- 3.1.4.Способы реализации интерфейсов. Работа с объектами через интерфейсы.
- 3.1.5.Обработка события: автоматическое создание обработчиков.

Задание №1

Написать программу. Программа должна уметь выводить на экран список объектов, удалять объекты, добавлять объекты, изменять объекты. При решении задачи создать не менее четырех классов с атрибутами и методами.

Задание: В текстовом файле находятся записи об учебниках в библиотеке. В каждой строке записана информация об одной книге: номер книги, ФИО автора(-ов), название, год издания, количество страниц, ссылка для скачивания. Отдельной сущностью (классом) хранится информация о скачивании книги: номер книги, время скачивания, IP адрес, логин пользователя.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Программа написана верно. Программа выводит на экран список объектов, удаляет объекты, добавляет объекты, изменяет объекты. При решении задачи создано не менее четырех классов с атрибутами и методами.

4	Программа написана верно. Программа выводит на экран список объектов, удаляет объекты, добавляет объекты, изменяет объекты. При решении задачи создано менее четырех классов с атрибутами и методами.
3	Программа написана верно. Программа выводит на экран список объектов, удаляет объекты, добавляет объекты, изменяет объекты

Дидактическая единица: 1.8 объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения

Занятие(-я):

- 3.1.1. Модификаторы доступа к элементам класса. Переменные ссылочного типа и присваивание. Побочные эффекты множественных ссылок.
- 3.1.2. Методы классов. Вызов метода. Передача параметров по значению. Создание методов, возвращающих значения. Способы размещения методов. Конструкторы.
- 3.1.3. Синтаксис наследования. Скрытие и перекрытие методов.
- 3.1.4. Способы реализации интерфейсов. Работа с объектами через интерфейсы.
- 3.1.5. Обработка события: автоматическое создание обработчиков.

Задание №1

Дополнить предыдущее задание следующими условиями:

- 1. Вводимые пользователем данные должны проверяться на правильность.
- 2. Не менее одного раза используйте константу или поле для чтения.
- 3. Все атрибуты классов должны иметь уровень видимости private.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Студент выполнил верно все три условия.
4	Студент выполнил два условия.
3	Студент выполнил верно одно условие.

Дидактическая единица: 2.5 оформлять код программ в соответствии со стандартом кодирования

Занятие(-я):

- 2.2.5. Реализация простых циклических алгоритмов.
- 2.2.6. Решение задач на составление циклических алгоритмов.
- 2.2.7. Решение задач на составление циклических алгоритмов.
- 2.2.12. Решение задач с использованием функций работы со строками.
- 2.2.15. Реализация алгоритмов обработки текстовых данных.

2.2.16.Реализация сложных алгоритмов поиска и ввода-вывода.

Задание №1

исходные данные вводить с помощью ListVox. Вывод результата организовать в метку Label. Разработать метод класса Form, реализующий задание.

Задание: Дана строка символов. Сформировать из нее строку, в которой символы следуют в порядке возрастания их кодов. Добавлены комментарии.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Добавлены комментарии.
4	Задание выполнено верно.
3	В выполненном задании имеются незначительные ошибки.

Дидактическая единица: 2.6 выполнять проверку, отладку кода программы

Занятие(-я):

2.2.6.Решение задач на составление циклических алгоритмов.

2.2.12.Решение задач с использованием функций работы со строками.

2.2.15.Реализация алгоритмов обработки текстовых данных.

2.2.16.Реализация сложных алгоритмов поиска и ввода-вывода.

Задание №1

Создать компьютерную игру "Сапер". Спроектировать основные объекты и сцену. Спроектировать логику поведения объектов. Спроектировать пользовательский интерфейс.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Прослеживается логика поведения объектов. Спроектирован удобный пользовательский интерфейс.
4	Задание выполнено верно. Прослеживается логика поведения объектов.
3	Задание выполнено верно. Нарушена логика поведения объектов.

2.6 Текущий контроль (ТК) № 6

Тема занятия: 3.1.10.Использование интерфейсов при реализации иерархии классов.

Метод и форма контроля: Практическая работа (Опрос)

Вид контроля: Практическая работа с использованием ИКТ

Дидактическая единица: 1.3 понятие системы программирования

Занятие(-я):

2.1.1.Классификация и генеалогия актуальных языков программирования. Понятие

системы программирования.

2.1.3. Основные элементы языка. Структура типовой программы. Особенности актуальных сред программирования.

2.1.4. Изучение инструментария среды программирования.

2.1.5. Подготовка структуры программы в среде программирования.

2.2.1. Методы реализации типовых алгоритмов. Переменные: определение, правила именования. Типы данных: значимые и ссылочные. Объявление и инициализация переменных. Область действия и время существования переменных. Константы: определение, виды и правила записи в программе.

2.2.4. Операторы цикла. Стандартные операции при работе с циклическими алгоритмами. Принудительный выход из цикла.

2.2.13. Управляющие структуры. Понятие потока. Механизм буферизации. Классы памяти. Доступ к файлам.

Задание №1

Выполнить следующее задание: В старояпонском календаре был принят двенадцатилетний цикл. Годы внутри цикла носили названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, петуха, собаки и свиньи. Написать программу, которая позволяет ввести номер года и печатает его название по старояпонскому календарю. Вводится год, и программа отображает информацию о животном-символе и соответствующую картинку (компонент pictureBox). Свойство ImageLocation хранит полный путь к файлу картинки. Справка: 1996 г. – год крысы – начало очередного цикла

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно.
4	В выполненном задании допущено одна ошибка.
3	В выполненном задании допущено две ошибки.

Дидактическая единица: 1.6 операторы и операции, управляющие структуры, структуры данных, классы памяти

Занятие(-я):

3.1.6. Создание классов для обработки массива данных.

3.1.7. Создание классов для обработки массива данных.

3.1.8. Создание классов для вычисления математических выражений.

3.1.9. Разработка проектов с обработкой событий.

Задание №1

Выполнить задания на двумерный массив: Точки на плоскости заданы координатами x и y , которые хранятся в двумерном массиве. Найти пару самых удаленных друг от друга точек. Имеется наличие комментариев, где это необходимо.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно, присутствуют комментарии.
4	Задание выполнено верно.
3	Задание выполнено с ошибками.

Дидактическая единица: 1.7 понятие подпрограммы, библиотеки подпрограмм
Занятие(-я):

2.2.14. Понятие подпрограммы, библиотеки подпрограмм. Библиотеки среды разработки.

2.2.15. Реализация алгоритмов обработки текстовых данных.

2.2.16. Реализация сложных алгоритмов поиска и ввода-вывода.

3.1.1. Модификаторы доступа к элементам класса. Переменные ссылочного типа и присваивание. Побочные эффекты множественных ссылок.

3.1.2. Методы классов. Вызов метода. Передача параметров по значению. Создание методов, возвращающих значения. Способы размещения методов. Конструкторы.

3.1.3. Синтаксис наследования. Скрытие и перекрытие методов.

3.1.4. Способы реализации интерфейсов. Работа с объектами через интерфейсы.

3.1.5. Обработка события: автоматическое создание обработчиков.

3.1.6. Создание классов для обработки массива данных.

3.1.7. Создание классов для обработки массива данных.

3.1.8. Создание классов для вычисления математических выражений.

3.1.9. Разработка проектов с обработкой событий.

Задание №1

Создать приложение «Проверка знаний по английскому языку», в котором учащийся выбирает тему для тестирования из списка тем. После выбора темы загружаются два неупорядоченных списка с английскими и русскими словами. Тестирование заключается в том, чтобы найти соответствие русских и английских слов. Подсчитывается количество правильных и неверных ответов.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Подсчитывается количество правильных и неверных ответов.
4	Задание выполнено верно. Подсчитывается количество верных ответов.
3	Задание выполнено верно. Подсчет ответов не реализован.

Дидактическая единица: 1.8 объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и

объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения

Занятие(-я):

3.1.6.Создание классов для обработки массива данных.

3.1.7.Создание классов для обработки массива данных.

3.1.8.Создание классов для вычисления математических выражений.

3.1.9.Разработка проектов с обработкой событий.

Задание №1

1. Создать публичный класс, задания (Учет продажи путевок, Создать родительский класс «Путевки» (код путевки, фамилия клиента, название пансионата, номер, вид жилья, дата заезда, дата выезда, количество человек, цена) и дочерние классы: «Зарубежные путевки» (загран паспорт, страховка); «Санатории» (мед.полис, диагноз, направление); «Детские оздоровительные» (возраст ребенка, свидетельство о рождении, пол). Реализовать класс для хранения списка путевок с методом добавления путевки и методом печати списка путевок).

2. Создать модель данных таблицы. Для этого создать класс, расширяющий абстрактный класс `AbstractTableModel`. Создать в нем объект коллекции типа `T`, соответствующий варианту задания.

Переопределить методы:

- *public Class<?> getColumnClass(int columnIndex)*
- *public int getColumnCount()*

- *public String getColumnName(int columnIndex)*
- *public Object getValueAt(int rowIndex, int columnIndex)*
- *public boolean isCellEditable(int rowIndex, int columnIndex)*

- *public void setValueAt(Object value, int rowIndex, int columnIndex)*

Определить методы:

- *public void addRow(<объект>) – добавления элемента (строки)*

- *public void deleteRow(String Поле_1) – удаления элемента по значению поля 1*

- *public void updateRow(int row, <объект>) – изменения элемента заданной строки)*

3. Для реализации окна приложения реализовать дочерний класс `JFrame`.

4. Для представления таблицы с данными использовать компонент класса `JTable`,

разместив его в контейнере JScrollPane (для возможности добавления полос прокрутки).

5. Поля ввода для добавления и редактирования данных реализовать текстовыми компонентами JTextField. Каждое поле снабдить подписью при помощи компонентов JLabel.

6. Для выполнения действий открытия файла, добавления, изменения, удаления записи, сортировки и сохранения файла реализовать соответствующие кнопки, с использованием компонентов JButton и добавлением ActionListener. Реализовать загрузку записи в поля для редактирования при щелчке по строке таблицы.

7. Для выбора файла при открытии и сохранении использовать компонент JFileChooser.

Учет продажи путевок, Создать родительский класс «Путевки» (код путевки, фамилия клиента, название пансионата, номер, вид жилья, дата заезда, дата выезда, количество человек, цена) и дочерние классы:

- «Зарубежные путевки» (загран паспорт, страховка);
- «Санатории» (мед.полис, диагноз, направление);
- «Детские оздоровительные» (возраст ребенка, свидетельство о рождении, пол).

Реализовать класс для хранения списка путевок с методом добавления путевки и методом печати списка путевок.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Учтены все требования и рекомендации.
4	Задание выполнено верно. Учтены 5 требований.
3	Задание выполнено верно. Учтены 3 требования.

Дидактическая единица: 2.4 использовать средства проектирования для создания и графического отображения алгоритмов

Занятие(-я):

2.2.12.Решение задач с использованием функций работы со строками.

3.1.8.Создание классов для вычисления математических выражений.

3.1.9.Разработка проектов с обработкой событий.

Задание №1

Создание блок-схем алгоритмов в пакете ms Visio следующей задачи: Дана целочисленная матрица $A(NM)$. Найдите номер первой из ее строк, которые начинаются с K положительных чисел подряд.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Верно построена блок-схема.
4	Блок-схема построена с одной ошибкой.
3	Блок-схема построена с двумя ошибками.

3. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

№ семестра	Вид промежуточной аттестации
5	Дифференцированный зачет

Дифференцированный зачет может быть выставлен автоматически по результатам текущих контролей
Текущий контроль №1
Текущий контроль №2
Текущий контроль №3
Текущий контроль №4
Текущий контроль №5
Текущий контроль №6

Метод и форма контроля: Практическая работа (Опрос)

Вид контроля: По выбору выполнить 1 теоретическое задание и 1 практическое задание

Дидактическая единица для контроля:

1.1 понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции

Задание №1

Ответить на вопросы:

Вопрос 1 Для решения любой задачи с помощью компьютера необходимо выполнить следующие этапы:

Установите правильную последовательность этапов.

- | | |
|---|-------------------------------|
| 1 | принятие решений; |
| 2 | программирование; |
| 3 | математическое моделирование; |
| 4 | алгоритмизация задач; |
| 5 | постановка задачи; |
| 6 | анализ результатов. |

Вопрос 2 *Соотнесите свойства алгоритма с их описанием:*

Результативность	- алгоритм должен приводить к решению задачи обязательно за конечное время;
------------------	---

Конечность	- неоднозначность толкования алгоритма недопустима;
Эффективность	- алгоритм должен обеспечить выдачу результата решения задачи на печать, на экран монитора или в файл;
Массовость	- правильный результат по алгоритму получен для одних исходных данных, то правильный результат по этому же алгоритму должен быть получен и для других исходных данных, допустимых в данной задаче;
Определенность	- позволяет решить задачу за приемлемое для разработчика время;

Вопрос 3 Выберите тип алгоритма, описанного ниже:

Набор команд (указаний), выполняемых последовательно друг за другом

1. линейный;
2. разветвляющийся;
3. циклический.

Вопрос 4 Выберите тип алгоритма, описанного ниже:

Алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения

1. линейный;
2. разветвляющийся;
3. циклический.

Вопрос 5 Выберите тип алгоритма, описанного ниже:

Алгоритм, предусматривающий многократное повторение одного и того же действия над новыми исходными данными

1. линейный;
2. разветвляющийся;
3. циклический.

<i>Оценка</i>	<i>Показатели оценки</i>
3	Даны правильные ответы на 3 вопроса
4	Даны правильные ответы на 4 вопроса
5	Даны правильные ответы на 5 вопросов

Задание №2

Ответить на вопросы:

1. Объясните на примере принцип работы полной и краткой формы условного оператора.
2. как показать условный оператор на схеме алгоритма? Приведите примеры.
3. Можно ли использовать условный оператор внутри другого условного оператора (например, в ветви else)?

<i>Оценка</i>	<i>Показатели оценки</i>
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

Задание №3

Ответить на вопросы:

1. Из чего состоит итерация для оператора while? Объясните на примере.
2. Объясните механизм работы оператора do-while на примерах.
3. Запишите в общем виде оператор цикла for. Какие ключевые слова при этом используются.

<i>Оценка</i>	<i>Показатели оценки</i>
3	Даны ответы на 2 вопроса, примеры не приведены
4	Даны ответы на все вопросы, примеры не приведены
5	Даны ответы на все вопросы, приведены примеры

Задание №4 (из текущего контроля)

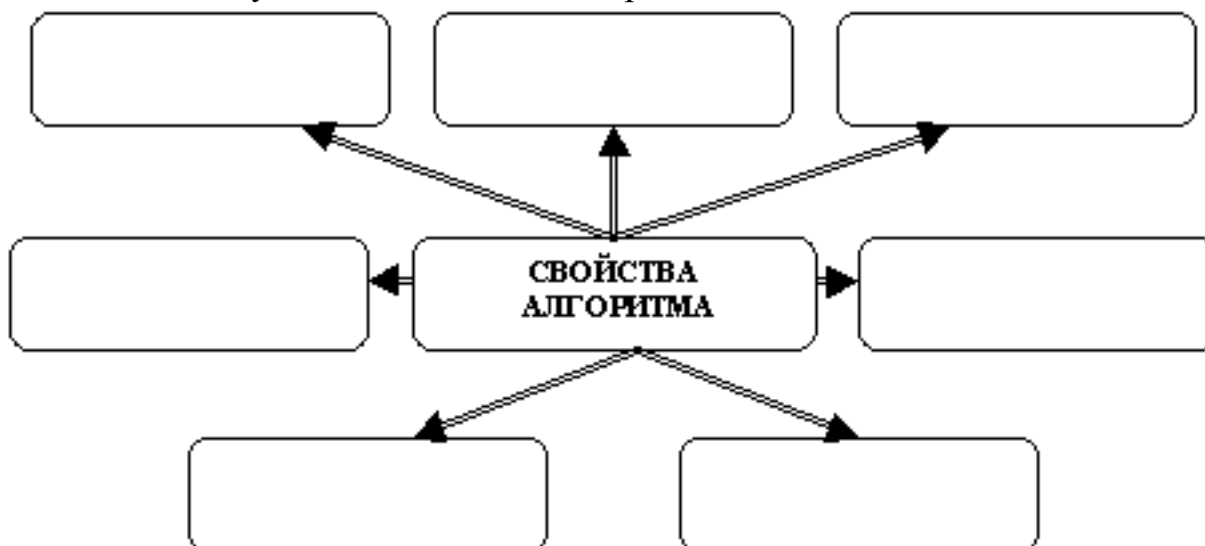
Дать развернутые ответы на следующие вопросы:

1. Дайте определение понятию "Алгоритм".
2. Перечислите и объясните свойства алгоритма.
3. Назовите формы записи алгоритмов.
4. Назовите и изобразите 5 основных элементов блок-схем.

<i>Оценка</i>	<i>Показатели оценки</i>
5	На все вопросы даны правильные ответы.
4	На один из вопросов не дан ответ / допущены ошибки в формулировке ответа.
3	На два вопроса не даны ответы / на один вопрос не дан ответ и допущены ошибки в формулировке ответа.

Задание №5 (из текущего контроля)

Заполните схему, вписав свойства алгоритма:



<i>Оценка</i>	<i>Показатели оценки</i>
5	Студент верно записал все свойства алгоритма.
4	Студент записал все свойства алгоритма, кроме одного.
3	Студент записал все свойства алгоритма, кроме трех.

Дидактическая единица для контроля:

1.2 классификация языков программирования

Задание №1 (из текущего контроля)

Вопрос 1. Как называется набор правил описания последовательностей символов

составляющих программу и какое вычисление производит программа?

- Язык программирования.
- Программа.
- Синтаксис программы.

Вопрос 2. Как называется последовательность символов определяющих вычисления? Данную запись на языках программирования называют исходным кодом.

- Программа.
- Синтаксис программы.
- Язык программирования.

Вопрос 3. Приведите классификация языков программирования в правильном соответствии

Символьные языки, -
максимально
приближенные к
машинным командам
конкретных ПЭВМ. Языки
низкого уровня
языки, ни в коей мере не -
ориентированные на
конкретные ПЭВМ. Языки
высокого уровня

машинно-
ориентированные

машинно-независимые
языки

<i>Оценка</i>	<i>Показатели оценки</i>
5	Правильно отвечено на 3 вопроса.
4	Правильно отвечено на 2 вопроса.
3	Правильно отвечено на 1 вопрос.

Дидактическая единица для контроля:

1.3 понятие системы программирования

Задание №1 (из текущего контроля)

Выполнить следующее задание: В старояпонском календаре был принят двенадцатилетний цикл. Годы внутри цикла носили названия животных: крысы,

коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, петуха, собаки и свиньи. Написать программу, которая позволяет ввести номер года и печатает его название по старояпонскому календарю. Вводится год, и программа отображает информацию о животном-символе и соответствующую картинку (компонент pictureBox). Свойство ImageLocation хранит полный путь к файлу картинки.
Справка: 1996 г. – год крысы – начало очередного цикла

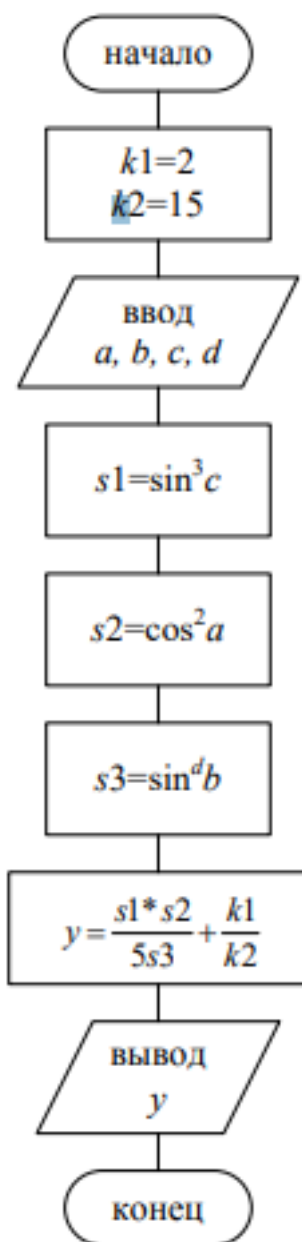
<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно.
4	В выполненном задании допущено одна ошибка.
3	В выполненном задании допущено две ошибки.

Дидактическая единица для контроля:

1.4 основные элементы языка, структура программы

Задание №1 (из текущего контроля)

Вычислить значение функции $15 \cdot 2 \cdot 5 \sin \sin \cos 3 \cdot 2 + x = b \cdot c \cdot a \cdot y \cdot d$ при $A = 9,5$; $B = 1,365$; $C = 6,5$; $D = 5$. Использовать два варианта ввода исходных данных и вывода результатов: возможности библиотеки функций языка C и библиотеки классов языка C#. Схема программы для задания представлена на рисунке



<i>Оценка</i>	<i>Показатели оценки</i>
5	Программа полностью соответствует алгоритму.
4	Программа соответствует алгоритму частично.
3	Программа не соответствует алгоритму.

Дидактическая единица для контроля:

1.5 методы реализации типовых алгоритмов

Задание №1 (из текущего контроля)

Составить алгоритм следующих задач:

1. Вычислить периметр и площадь прямоугольного треугольника по заданным длинам двух катетов a и b .

2. Найти максимальное значение из трех различных целых чисел, введенных с клавиатуры.
3. Дано двузначное число. Определить: какая из его цифр больше: первая или вторая.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм трех задач составлен верно.
4	Алгоритм двух задач составлен верно.
3	Алгоритм одной задачи составлен верно.

Дидактическая единица для контроля:

1.6 операторы и операции, управляющие структуры, структуры данных, классы памяти

Задание №1 (из текущего контроля)

Написать программу. Программа должна уметь выводить на экран список объектов, удалять объекты, добавлять объекты, изменять объекты. При решении задачи создать не менее четырех классов с атрибутами и методами.

Задание: В текстовом файле находятся записи об учебниках в библиотеке. В каждой строке записана информация об одной книге: номер книги, ФИО автора(-ов), название, год издания, количество страниц, ссылка для скачивания. Отдельной сущностью (классом) хранится информация о скачивании книги: номер книги, время скачивания, IP адрес, логин пользователя.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Программа написана верно. Программа выводит на экран список объектов, удаляет объекты, добавляет объекты, изменяет объекты. При решении задачи создано не менее четырех классов с атрибутами и методами.
4	Программа написана верно. Программа выводит на экран список объектов, удаляет объекты, добавляет объекты, изменяет объекты. При решении задачи создано менее четырех классов с атрибутами и методами.
3	Программа написана верно. Программа выводит на экран список объектов, удаляет объекты, добавляет объекты, изменяет объекты

Дидактическая единица для контроля:

1.7 понятие подпрограммы, библиотеки подпрограмм

Задание №1 (из текущего контроля)

Создать приложение «Проверка знаний по английскому языку», в котором учащийся выбирает тему для тестирования из списка тем. После выбора темы

загружаются два неупорядоченных списка с английскими и русскими словами. Тестирование заключается в том, чтобы найти соответствие русских и английских слов. Подсчитывается количество правильных и неверных ответов.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Подсчитывается количество правильных и неверных ответов.
4	Задание выполнено верно. Подсчитывается количество верных ответов.
3	Задание выполнено верно. Подсчет ответов не реализован.

Дидактическая единица для контроля:

1.8 объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения

Задание №1 (из текущего контроля)

Дополнить предыдущее задание следующими условиями:

1. Вводимые пользователем данные должны проверяться на правильность.
2. Не менее одного раза используйте константу или поле для чтения.
3. Все атрибуты классов должны иметь уровень видимости private.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Студент выполнил верно все три условия.
4	Студент выполнил два условия.
3	Студент выполнил верно одно условие.

Задание №2 (из текущего контроля)

1. Создать публичный класс, задания (Учет продажи путевок, Создать родительский класс «Путевки» (код путевки, фамилия клиента, название пансионата, номер, вид жилья, дата заезда, дата выезда, количество человек, цена) и дочерние классы: «Зарубежные путевки» (загран паспорт, страховка); «Санатории» (мед.полис, диагноз, направление); «Детские оздоровительные» (возраст ребенка, свидетельство о рождении, пол). Реализовать класс для хранения списка путевок с методом добавления путевки и методом печати списка путевок).

2. Создать модель данных таблицы. Для этого создать класс, расширяющий абстрактный класс AbstractTableModel. Создать в нем объект коллекции типа T,

соответствующий варианту задания.

Переопределить методы:

- *public Class<?> getColumnClass(int columnIndex)*
- *public int getColumnCount()*

- *public String getColumnName(int columnIndex)*
- *public Object getValueAt(int rowIndex, int columnIndex)*
- *public boolean isCellEditable(int rowIndex, int columnIndex)*

- *public void setValueAt(Object value, int rowIndex, int columnIndex)*

Определить методы:

- *public void addRow(<объект>) – добавления элемента (строки)*

- *public void deleteRow(String Поле_1) – удаления элемента по значению поля 1*

- *public void updateRow(int row, <объект>) – изменения элемента заданной строки)*

3. Для реализации окна приложения реализовать дочерний класс JFrame.

4. Для представления таблицы с данными использовать компонент класса JTable, разместив его в контейнере JScrollPane (для возможности добавления полос прокрутки).

5. Поля ввода для добавления и редактирования данных реализовать текстовыми компонентами JTextField. Каждое поле снабдить подписью при помощи компонентов JLabel.

6. Для выполнения действий открытия файла, добавления, изменения, удаления записи, сортировки и сохранения файла реализовать соответствующие кнопки, с использованием компонентов JButton и добавлением ActionListener. Реализовать загрузку записи в поля для редактирования при щелчке по строке таблицы.

7. Для выбора файла при открытии и сохранении использовать компонент

JFileChooser.

Учет продажи путевок, Создать родительский класс «Путевки» (код путевки, фамилия клиента, название пансионата, номер, вид жилья, дата заезда, дата выезда, количество человек, цена) и дочерние классы:

- «Зарубежные путевки» (загран паспорт, страховка);
- «Санатории» (мед.полис, диагноз, направление);
- «Детские оздоровительные» (возраст ребенка, свидетельство о рождении, пол).

Реализовать класс для хранения списка путевок с методом добавления путевки и методом печати списка путевок.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Учтены все требования и рекомендации.
4	Задание выполнено верно. Учтены 5 требований.
3	Задание выполнено верно. Учтены 3 требования.

Дидактическая единица для контроля:

2.1 разрабатывать и анализировать алгоритмы для решения поставленных задач

Задание №1 (из текущего контроля)

Построить алгоритм решения задачи в виде блок-схемы.

Задача: Найти среднее арифметическое чисел, введенных с клавиатуры. Признак конца ввода - ноль. Определить количество введенных чисел.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм составлен без ошибок.
4	Алгоритм составлен с одной ошибкой.
3	Алгоритм составлен с двумя ошибками.

Дидактическая единица для контроля:

2.2 определять сложность алгоритмов

Задание №1 (из текущего контроля)

Построить алгоритм решения задачи в виде блок-схемы:

1. С клавиатуры ввести последовательность символов. Написать функцию, вычисляющую длину этой последовательности.

2. Сформировать матрицу {7,8}. Вывести матрицу. найти и вывести значение минимального элемента. вывести координаты всех минимальных элементов матрицы и их количество.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм составлен без ошибок.

4	Алгоритм составлен с одной ошибкой.
3	Алгоритм составлен с двумя ошибками.

Дидактическая единица для контроля:

2.3 реализовывать типовые алгоритмы в виде программ на актуальных языках программирования

Задание №1 (из текущего контроля)

Написать программу: С помощью генератора случайных чисел сформировать массив D из 18 элементов. Вывести его на экран. Найти сумму положительных элементов массива, произведение отрицательных и количество нулей.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Программа написана верно.
4	Программа написана с одной ошибкой.
3	Программа написана с двумя ошибками.

Задание №2

Разработайте алгоритм для одной из задачи с использованием программы для графического отображения алгоритмов:

В 1202 г. итальянский математик Леонард Фибоначчи подсчитывал, на сколько увеличивается число кроликов в хозяйстве каждый год. При этом он получил последовательность такого вида: 1, 1, 2, 3, 5, 8, 13, 21, 34 Написать программу, которая для заданного числа A выводит N членов последовательности Фибоначчи.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм составлен без ошибок
4	Алгоритм составлен без ошибок
3	Алгоритм составлен с двумя ошибками

Дидактическая единица для контроля:

2.4 использовать средства проектирования для создания и графического отображения алгоритмов

Задание №1 (из текущего контроля)

Создание блок-схем алгоритмов в пакете ms Visio следующей задачи: Дана целочисленная матрица A(NM). Найдите номер первой из ее строк, которые начинаются с K положительных чисел подряд.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Верно построена блок-схема.
4	Блок-схема построена с одной ошибкой.
3	Блок-схема построена с двумя ошибками.

Задание №2

Разработайте алгоритм для одной из задачи с использованием программы для графического отображения алгоритмов:

Ввести два целых числа. Вывести в порядке убывания все числа, лежащие между ними, и количество этих чисел. Каждое третье число не печатать и не учитывать

<i>Оценка</i>	<i>Показатели оценки</i>
5	Алгоритм составлен без ошибок
4	Алгоритм составлен с одной ошибкой
3	Алгоритм составлен с двумя ошибками

Дидактическая единица для контроля:

2.5 оформлять код программ в соответствии со стандартом кодирования

Задание №1 (из текущего контроля)

исходные данные вводить с помощью ListBox. Вывод результата организовать в метку Label. Разработать метод класса Form, реализующий задание.

Задание: Дана строка символов. Сформировать из нее строку, в которой символы следуют в порядке возрастания их кодов. Добавлены комментарии.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Добавлены комментарии.
4	Задание выполнено верно.
3	В выполненном задании имеются незначительные ошибки.

Задание №2

Оформите представленный код программы в соответствии со стандартом кодирования:

Вершины треугольника заданы массивом точек. Точки являются объектом класса CPoint. Написать функцию, вычисляющую площадь треугольника по формуле Герона:

```

#include #include class CPoint
{ private: int x, y; public:
CPoint(int _x, int _y)
(x = _x; y = _J/;}
CPoint()
{x = 0; y = 0; } void SetX(int _x)
{x = _x;}
void SetY(int _y)
(Y = _Y; > int GetX()
{ return x; } int GetY() { return y; } };
float length(CPoint pi, CPoint p2)
{
int x1, x2, y1, y2, L; x1 = pi.GetX(); y1 = pi.GetY(); x2 = p2.GetX(); y2 = p2.GetY();
L = sqrt(pow(x2-x1, 2.0) + pow(y2-y1, 2.0)); return L;
float square(CPoint *mas)
{float L1, L2, L3, p;
L1 = length(mas[0], mas[1]);
L2 = length(mas[1], mas[2]);
L3 = length(mas[2], mas[0]); p = (L1 + L2 + L3)/2.0; if (p*(p-L1)*(p-L2)*(p-L3)>=0)
return sqrt(p*(p - L1)*(p - L2)*(p - L3)); else (printf("Треугольник построить нельзя! ");
return -1;
}} int main(){
CPoint mas [3]; int x, y;
for (int i = 0; i<3; i++){printf("First point:"); scanf("%d%d", &x, &y); mas [i] .SetX(x);
mas[i].SetY(y);}
printf("S = %3.2f ", square(mas)); return 0;}

```

<i>Оценка</i>	<i>Показатели оценки</i>
5	программа оформлена частично в соответствии со стандартом кодирования
4	программа оформлена в соответствии со стандартом кодирования, с небольшими недочетами
3	программа оформлена частично в соответствии со стандартом кодирования программа оформлена в соответствии со стандартом кодирования, с небольшими недочетами

Дидактическая единица для контроля:

2.6 выполнять проверку, отладку кода программы

Задание №1

Выполните проверку и отладку следующего кода программы

Написать класс «герой». Члены класса: имя героя, его возраст. Написать класс «параметры». Члены класса: сила, защита, скорость. На базе этих двух классов написать класс «воин», включающий в себя все перечисленные члены.

Иерархия классов будет выглядеть следующим образом:

```
#include
#include
class Hero //начало определения 1-го базового класса Hero
{
protected:
char name [32 ] ; //защищенная член-переменная name класса Hero
int age; //защищенная член-переменная age класса Hero
public:
Hero () //определение конструктора класса Hero без параметров
{
strcpy(name, "NoName");
//копирование в член-переменную name текста NoName age = 0; //присвоить члену-
переменной age значения 0
}
Hero(char* name, int age)
//определение конструктора класса Hero с двумя параметрами
{
strcpy (this ->name, name) ; /*копирование члена-переменной
name (this->name,) значения параметра name V
this->age = age;
//присвоить члену-переменной age (this->age^ значение параметра age
}
void info () //определения функции-члена info
{
printf("Hero:%s, %d ", name, age);
//вывод на экран названия класса и значений членов-переменных name и age
}
}; //конец определения базового класса Hero
class Parametry //начало определения базового класса Parametry
{
protected:
int si la; //защищенная член-переменная si la класса Parametry
int zashita;
//защищенная член-переменная zashita класса Parametry
int skorost;
```

```

//защищенная член-переменная skorost класса Parametr
public:
Parametry(int _sila = 0, int _zashita = 0, int _skoros = 0)
//конструктор класса Parametry со значениями поумолчаниюю
{
sila = _sila; zashita = _zashita; skorost = _skorost;
}
void info() //определение члена-функции info класса Parametry
printf("Parametry: %d, %d, %d ", sila, zashita,
skorost) ; /*вывод на экран названия класса и значений
членов-переменных sila, zashita, skorost*/
}
}; //конец определения класса Parametry
class Warrior : public Hero, Parametry
//начало определ. класса Warrior наследника классов Hero, Parametry { public:
/*определение конструктора класса Warrior инициализацией «родительских»
конструкторов классов Hero и Parametry*/
Warrior(char* _name, int _age, int _sila, int _zashita, int _skorost): Hero(_name, _age),
Parametry(_sila, _zashita, _skoros)
{ }
void info() //определение члена-функции info класса Warrior
{
printf("Warrior: ");
//вывод на экран названия класса Warrior Hero : : info () ; //вызов функции-члена info
класса Hero
Parametry::info();
//вызов функции-члена info класса Parametry
}
}; //конец определения класса Warrior
int main()
Warrior w("Ivan", 30, 100, 300, 15); w.info(); return 0;
}

```

Оценка	Показатели оценки
3	проверка и отладка выполнена, программа работает с ошибками
4	проверка и отладка выполнена, программа работает с небольшими недочетами
5	проверка и отладка выполнена, программа работает верно

Задание №2 (из текущего контроля)

Создать компьютерную игру "Сапер". Спроектировать основные объекты и сцену. Спроектировать логику поведения объектов. Спроектировать пользовательский интерфейс.

<i>Оценка</i>	<i>Показатели оценки</i>
5	Задание выполнено верно. Прослеживается логика поведения объектов. Спроектирован удобный пользовательский интерфейс.
4	Задание выполнено верно. Прослеживается логика поведения объектов.
3	Задание выполнено верно. Нарушена логика поведения объектов.