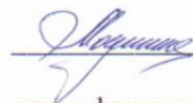


УТВЕРЖДАЮ



В. И. Модинец

декан факультета повышения
квалификации преподавателей,
руководитель мероприятия

**Методическое пособие по проведению обучающего модуля
«Программирование»**

Пояснительная записка

Модуль «Программирование» разработан с ориентацией на соответствующий учебный план (профессиональный цикл) и рабочую программу дисциплины «Основы программирования», входящих в основную образовательную программу высшего образования по направлению подготовки бакалавров 010300 Фундаментальные информатика и информационные технологии.

Мы живем в обществе, где каждый использует мобильный телефон или компьютер, а у большинства имеются оба устройства. Современные дети живут в совершенно ином мире, чем их родители, когда были детьми. Сегодняшний мир полон веб-сервисов — ютуб, фейсбук и вконтакте стали ключевой частью повседневной жизни детей. Даже игрушки сегодня стали цифровыми и многие являются программируемыми, такие как Лего нового поколения LeapFrogs, с датчиками. Но одно дело знать, как использовать эти программы, а другое понять, как устроена логика их работы. Знание основ программирования необходимо для понимания современного цифрового мира и законов жизни в нем. Количество технологий и наша зависимость от них только увеличивается, именно поэтому сегодня обучающиеся должны быть в состоянии не только пользоваться этими технологиями, но и понять как ими управлять.

У каждого есть идеи. К сожалению, лишь немногие могут воплотить их в жизнь. Без надлежащего исполнения, идеи никогда не будут достигать своего потенциала. Что отличает тех, у кого просто есть идея, от тех, кто воплощает свои идеи в реальность? Способность программировать.

На самом деле, это достаточно просто. Мы получаем немедленную обратную связь. Например, если ребенок пишет программу, чтобы окрасить фигуру в конкретный цвет, затем сразу видит результат, который хотел получить, значит его код работает. Хорошая работа!

Помимо этого, требования высших учебных заведений к выпускникам школ существенно возросли и расширились. Это обстоятельство вынуждает особенно тщательно согласовывать разработку программы преподавания дисциплины Информатика в школе с дальнейшим изучением соответствующего материала в вузе.

Модуль «Основы программирования» базируется на современных системах и языках программирования, решении практических задач. Значительное внимание уделяется фундаментальным вопросам технического и технологического обеспечения информатики, логическим и арифметическим основам компьютера.

Содержание курса сочетает в себе три существенных сейчас основных подхода в обучении информатики в школе и отражает важнейшие аспекты ее образовательной значимости:

- «пользовательский» аспект, связанный с формированием компьютерной грамотности, информационной культуры, подготовкой школьников к практической деятельности в условиях широкого использования информационных технологий. Такое обучение целесообразно вводить как можно раньше, чтобы учащиеся могли использовать различные доступные их возрасту программные продукты, применяя компьютер в качестве инструмента для своих целей;
- алгоритмический (программистский) аспект, связанный в большей мере с развитием мышления учащихся;
- кибернетический аспект, связанный с формированием мировоззренческих представлений о роли информации в управлении, специфики самоуправляющихся систем, общих закономерностях информационных процессов в системах разной природы.

Курс в целом охватывает следующие группы вопросов:

- Вопросы, связанные с пониманием сущности информационных процессов, информационных основ процессов управления в системах различной природы и представлением о передаче информации, канале передачи информации, количестве информации, способах представления информации для формального исполнителя (информационный аспект);
- Методы и средства формализованного описания действий исполнителя (алгоритмический аспект);
- Вопросы, связанные с выбором исполнителя для решения задачи, анализом его свойств, возможностей и эффективности его применения для решения данной задачи;
- Вопросы, связанные с адекватным описанием реальных объектов и явлений для их использования с помощью ЭВМ, проведение компьютерного эксперимента (моделирование).

В процессе обучения заложены следующие необходимые умения и навыки:

- умение планировать структуру действий, необходимых для достижения заданной цели при помощи фиксированного набора средств;
- умение организовывать поиск информации, необходимой для решения поставленной задачи;
- умение строить информационные структуры (модели) для описания объектов и систем;
- технические навыки работы с компьютером и его периферийными устройствами.

Именно при формировании таких навыков и умений можно будет говорить о высокой эффективности использования компьютеров и достижении учащихся не только уровня знаний и умений, заложенных в «Обязательный минимум содержания образования», но и достаточно более высокого уровня, необходимого в дальнейшем обучении.

Курс рассчитан на 16 часов и предусматривает выполнение 4 практических работ по различным темам.

При изучении языка программирования следует стремиться к использованию формального описания конструкций языка программирования (синтаксис и семантику), а при решении задач использовать только изученные элементы. Это способствует не только глубокому пониманию учебного материала, но и дает возможность использовать изученный материал в дальнейшем.

При обучении программированию следует добиваться определенного стиля оформления текстов программ.

1. Цели освоения модуля

Цель настоящего модуля – на этапе обучения в старших классах дать основные сведения о языках программирования, средствах языков, структуре программ и программировать различные задачи с использованием различных циклов.

Задачи модуля:

- раскрыть сущность типовых алгоритмов: ввод-вывод данных, использование циклов, работа с массивами;
- ознакомить со структурированными типами данных;
- дать навыки структурного программирования как метода, предполагающего создание понятных программ, обладающих свойствами модульности;
- изучить всевозможные методы решения задач, реализуемых на различных языках программирования;
- научиться основам алгоритмического мышления;

2. Место учебной дисциплины в структуре основной образовательной программы:

Данный модуль относится к системе элективных курсов и является дополнением к предмету Информатика среднего общего образования. Изучение модуля дает основы для решения небольших задач, от тех, что встречаются в курсах изучения информатики и основ программирования в старших классах школы и на начальных курсах ВУЗов и до тех, с которыми могут столкнуться начинающие программисты на работе.

3. Основные образовательные результаты обучающегося, формируемые в результате освоения содержания модуля.

В рамках сложившейся системы образования содержание, предусмотренное настоящим модулем, осваивается частично в высших учебных заведениях, частично в системе послевузовского образования. При этом базовые представления о программировании можно передать обучающимся старших классов.

Компетенции обучающегося, формируемые в результате освоения модуля:

| Наименование компетенции | Результат освоения (знать, уметь, владеть) |
|---|--|
| способность планировать структуру действий, необходимых для достижения заданной цели при помощи фиксированного набора средств | <p>Знать:</p> <ul style="list-style-type: none"> – теоретические подходы к планированию структуры действий, необходимых для достижения заданной цели при помощи фиксированного набора средств; – основные способы поиска информации, необходимой для решения поставленной задачи; – особенности построения информационных структур (моделей) для описания объектов и систем; – возможности использования компьютера и его периферийных устройств. |
| способность организовывать поиск информации, необходимой для решения поставленной задачи | <p>Уметь:</p> <ul style="list-style-type: none"> • умение планировать структуру действий, необходимых для достижения заданной цели при помощи фиксированного набора средств; • умение организовывать поиск информации, необходимой для решения поставленной задачи; • умение строить информационные структуры (модели) для описания объектов и систем; |
| Способность строить информационные структуры | <ul style="list-style-type: none"> • технические навыки работы с компьютером и его периферийными устройствами. |

| | |
|--|--|
| (модели) для описания объектов и систем | |
| Владение техническими навыками работы с компьютером и его периферийными устройствами | |

4. Структура и содержание модуля

| Темы дисциплины | Основные понятия | Часы | Вид учебной работы |
|--|--|------|--|
| Основные понятия о языках программирования | Средства языка, алфавит языка программирования, служебные слова, структура программы, переменные и константы, типы данных | 2 | Лекция, практическое занятие Лек., ЛЗ |
| Встроенные функции в языки программирования | Форма записи встроенных функций; оператор ввода; оператор вывода; оператор присваивания; оператор комментария; блок-схема конструкции «Следование» | 2 | Практическое занятие ЛЗ, ПЗ |
| Условный оператор | Полный условный оператор, сокращенный условный оператор, вложенный условный | 2 | Практическое занятие ЛЗ, ПЗ |

| | | | |
|---|--|----|--------------------------------|
| | оператор, блок – схема | | |
| Циклы. | Тело цикла. Оператор цикла, Параметр цикла. Шаг цикла | 2 | Практическое занятие ЛЗ, ПЗ |
| Массивы | Массив, индекса, переменная с индексом, простая переменная, одномерные массивы. | 4 | Практическое занятие ЛЗ, ПЗ |
| Строковый, символьный типы данных | Символьные величины, стандартные функции, сравнение, конкатенация, копирование, удаление, замена (вставка), шифровка, дешифровка текста. | 2 | Практическое занятие ЛЗ, ПЗ |
| Процедуры и функции. Понятие подпрограмм. Механизм реализации подпрограмм с помощью процедур и функций | Рекурсия, рекурсивные алгоритмы, подпрограммы, процедуры, функции,. | 1 | Практическое занятие ЛЗ, ПЗ |
| Итоговое занятие | | 1 | СРС |
| ИТОГО: | | 16 | |

Тема 1. Основные понятия о языках программирования

История появления языков QBASIC и TURBO PASCAL. Языки программирования интерпретирующего и компилирующего типа. Основные средства языка. Алфавит

языка. Служебные слова. Структура программы. Переменные и константы. Типы данных. Присваивание. Технология разработки программного обеспечения. Система и языки программирования. Общая характеристика системы программирования. Система оперативной подсказки. Редактор исходного текста. Пример простой программы. Компиляция и отладка программы.

В результате изучения темы учащиеся должны знать:

- историю появления языков программирования QBASIC и TURBO PASCAL;
- отличия компилятора и интерпретатора;
- основные средства, алфавит языка программирования, служебные слова, структуру программы;
- команду присваивания;
- типы данных, переменные и константы;
- общую характеристику системы программирования.

В результате изучения темы учащиеся должны уметь:

- составлять простейшие программы с использованием команды присваивания;
- компилировать и отлаживать программу.

Тема 2. Встроенные функции в языки программирования

Запись на языке QBASIC функций $|X|$, $\sin x$, $\cos x$, $\operatorname{tg} x$, целая часть x , x^2 и т. д. Операторы INPUT, PRINT. Работа оператора присваивания. Выражение вида $N=N+1$. Блок-схема конструкции «Следование». Оператор комментария. Набор, отладка и запуск программы в среде программирования QBASIC.

В результате изучения темы учащиеся должны знать:

- форму записи встроенных функций;
- оператор ввода;
- оператор вывода;
- оператор присваивания;
- оператор комментария;
- блок-схему конструкции «Следование»

В результате изучения темы учащиеся должны уметь:

Составлять простейшие программы с использованием операторов присваивания, ввода-вывода информации.

Тема 3. Условный оператор

Блок-схема конструкции «Ветвление». Условный оператор. Служебные слова IF, THEN, ELSE. Работа полного условного оператора. Работа сокращенного условного оператора. Вложенные условные операторы. Блок – схема вложенных условных операторов. Конструкция вложенного условного оператора. Служебные слова IF, THEN, ELSE, END IF. Логические связки AND, OR, NOT. Программирование задач с использованием вложенных условных операторов и сложных условий. Блок-схема

оператора варианта. Структура оператора варианта. Служебные слова SELECT, CASE, END SELECT.

В результате изучения темы учащиеся должны знать:

- блок-схему полной формы условного оператора, блок-схему сокращенной формы условного оператора, блок-схему вложенного условного оператора, оператора варианта;
- запись условного оператора с помощью языка программирования;
- запись оператора варианта с помощью языка программирования.

В результате изучения темы учащиеся должны уметь:

- составлять программы с использованием полной и сокращенной формы условного оператора;
- составлять программы с использованием вложенных условных операторов;
- составлять программы, содержащие сложные условия;
- составлять программы с использованием оператора варианта.

Тема 4. Циклы

Блок-схема цикла с предусловием. Тело цикла. Оператор цикла с предусловием. Служебные слова WHILE, WEND. Работа оператора цикла с предусловием. Блок-схема цикла с постусловием. Тело цикла. Оператор цикла с постусловием. Служебные слова DO, LOOP, WHILE, UNTIL. Работа оператора цикла с постусловием. Блок-схема цикла с параметром. Тело цикла. Параметр цикла. Начальное и конечное значение параметра цикла. Шаг цикла. Оператор цикла с параметром. Служебные слова FOR, TO, NEXT. Работа оператора цикла с параметром. Конструкция вложенных циклов. Конструкция сочетания цикла и условного оператора.

В результате изучения темы учащиеся должны знать:

- блок-схему цикла с параметром, блок-схему цикла с предусловием, блок-схему цикла с постусловием;
- запись циклов с логическим условием;
- запись циклов с помощью языка программирования.

В результате изучения темы учащиеся должны уметь:

- программировать задачи с использованием циклов с параметром, с предусловием, с постусловием;
- программировать задачи с использованием циклов с логическим условием;
- программировать задачи с использованием вложенных циклов.

Тема 5. Массивы

Понятие массива. Понятие индекса. Переменная с индексом. Простая переменная. Одномерные массивы. Описание массива: DIM имя массива (n1 TO n2) AS тип элементов. Присвоение значений элементам массива. Способы задания одномерных массивов. Понятие матрицы. Двумерные массивы. Нумерация элементов

двумерного массива. Способы описания двумерного массива. Способы задания двумерных массивов. Проверка на четность. Подсчет количества элементов, отвечающих заданным условиям. Нахождение суммы, произведения и количества элементов массива, отвечающих заданным условиям. Максимальный и минимальный элементы. Сортировка элементов массива. Метод «пузырька». Оператор SWAP. Перестановка элементов массива. Поиск, подбор и группировка данных. Слияние и отбор данных в одномерных и двумерных массивах.

В результате изучения темы учащиеся должны знать:

- понятие массива, переменной с индексом, простой переменной;
- понятие одномерного массива;
- как происходит присвоение значений элементам одномерного массива;
- работу датчика случайных чисел;
- понятие двумерного массива;
- как происходит присвоение значений элементам двумерного массива;

В результате изучения темы учащиеся должны уметь:

- организовывать задание одномерного и двумерного массива;
- подсчитывать сумму, произведение и количество элементов одномерного и двумерного массивов, отвечающих заданным условиям;
- находить максимальный и минимальный элемент одномерного и двумерного массивов;
- осуществлять перестановку элементов одномерного и двумерного массивов;
- осуществлять слияние и отбор данных в одномерных и двумерных массивах;
- осуществлять поиск, подбор и группировку данных;
- осуществлять сортировку элементов массива.

Тема 6. Строковый, символьный типы данных.

Основные сведения о символьных величинах. Стандартные функции, процедуры для работы с символьными величинами: сравнение, конкатенация, копирование, удаление, замена (вставка), длина строки, подстрока числа и строки. Понятие шифровки, дешифровки текста. Способы шифровки текста.

В результате изучения темы учащиеся должны знать:

- основные сведения о символьных величинах;
- реализацию символьных величин с помощью языков программирования;
- стандартные функции для работы с символьными величинами: сравнение, конкатенацию, копирование, удаление, замену (вставку), длину строки, подстроку.

В результате изучения темы учащиеся должны уметь:

- программировать алгоритмы обработки текста;
- программировать задачи на поиск и подсчет (лингвистическая статистика).

Тема 7. Процедуры и функции. Понятие подпрограмм. Механизм реализации подпрограмм с помощью процедур и функций.

Понятие рекурсии. Рекурсивные алгоритмы. Задачи, сводимые к рекурсивным. Понятие подпрограммы, процедуры, функции, Описание процедуры. Оператор SUB. Описание функции. Оператор FUNCTION. Механизм реализации подпрограмм с помощью процедур и функций. Обмен информацией между основной программой и подпрограммой. Глобальные и локальные переменные.

В результате изучения темы учащиеся должны знать:

- понятие подпрограммы;
- понятие процедуры;
- оператор процедуры;
- понятие функции;
- оператор задания функции;
- как осуществляется обмен между основной программой и подпрограммой;
- понятие глобальной и локальной переменной;
- механизм реализации подпрограммы с помощью процедур и функций.

В результате изучения темы учащиеся должны уметь:

- уметь программировать задачи с использованием процедур и функций.

Работа с файлами

Понятие файла. Классификация файлов. Операции над файлами: открытие файла, чтение и запись обрабатываемых данных, закрытие файлов. Файл произвольного доступа. Операторы и функции работы с файлом произвольного доступа.

В результате изучения темы учащиеся должны знать:

- понятие файла: имя, расширение;
- типы файлов;
- операции над файлами;
- файлы с произвольным доступом.

В результате изучения темы учащиеся должны уметь:

- решать задачи на программирование с использованием файлов.

| Вид занятия | Используемые образовательные технологии |
|--------------------|---|
| Лек. | Активные (проблемные) лекции (лекция-беседа, лекция-конференция, лекция-презентация, лекция-дискуссия) |
| ЛЗ | Учебная дискуссия; имитационные упражнения; индивидуальная практическая работа (ИЗ) с интерактивной презентацией результатов, кейс-стади; ролевая игра; проектное занятие; тренинг, занятие с использованием практических заданий |

| | |
|-----|--|
| ПЗ | Подготовка к занятиям; индивидуальная практическая работа (работа с медиа-ресурсами, электронными носителями, источниками, базами) |
| СРС | Подготовка письменной работы |

Содержание учебного материала

Основные понятия о языках программирования

1. Понятие о языке программирования высокого и низкого уровня.

Цели: дать основные сведения о языках программирования, средствах языков, структуре программ.

Краткий обзор теоретического материала. История создания языков программирования Бейсик и Паскаль. Система и языки программирования. Технология разработки программного обеспечения. Общая характеристика системы программирования. Основные средства языка программирования. Алфавит языка. Служебные слова. Структура программы. Система оперативной подсказки. Редактор исходного текста. Пример простой программы. Компиляция и отладка программы.

Примеры возможных заданий.

Подготовить ответы на вопросы:

1. Для чего служит компьютер?
2. Что называется алгоритмом?
3. Что называют алфавитом языка?
4. Что такое интерпретация и компиляция?
5. Что представляет собой программа?
6. Что включает в себя алфавит языка QBASIC?
7. Выполнение творческого проекта (отчета) по теме «История языков программирования».

2. Переменные и константы. Описание типов.

Цели: дать основные сведения о константах, переменных; познакомить учащихся с различными типами данных.

Краткий обзор теоретического материала. Понятие переменной. Переменные числового типа: целые, длинные целые, обычной точности, двойной точности; переменные символьного типа: строка переменной длины, строка фиксированной длины; переменные пользовательского типа. Понятие константы.

Примеры возможных заданий.

Подготовить ответы на вопросы:

1. Чем характеризуется переменная?
2. В каком разделе происходит описание переменных?

3. Как описываются переменные?
4. Перечислите типы данных.
5. Какова структура программы?
6. С чего начинается основная программа?
7. Чем заканчивается программа?
8. Определите тип выражения: $1+0.0$; $20/4$.

Встроенные функции в языки программирования. Простейшие операторы языка QBASIC

1. Встроенные функции.

Цели: дать представления о встроенных функциях; научить учащихся использовать встроенные функции при записи математических выражений на языке программирования QBASIC.

Краткий обзор теоретического материала. Запись на языке QBASIC функций $|X|$, $\sin x$, $\cos x$, $\operatorname{tg} x$, целая часть x , x^2 и т. д.

Примеры возможных заданий.

Записать на языке QBASIC следующие выражения:

1. $Y=5\sin X$
2. $Z=14X^4 - 5X^3 + 11X - 17$
3. $X_1 = \frac{-b - \sqrt{D}}{2a}$
4. $D=b^2 - 4ac$

Сделать математическую запись следующих выражений:

1. $\operatorname{LOG}(X) - 32 / (\operatorname{ABS}(X) - 1)$
2. $\operatorname{SQR}(64)$

2. Операторы ввода и вывода информации. Оператор присваивания.

Цели: познакомить учащихся с операторами ввода – вывода информации, оператором присваивания; программировать простейшие задачи с использованием операторов ввода – вывода и присваивания.

Краткий обзор теоретического материала. Операторы INPUT, PRINT. Работа оператора присваивания. Выражение вида $N=N+1$. Блок-схема конструкции «Следование»

Примеры возможных заданий.

1. Вывести на экран приветствие: «Добрый день»
2. Вычислить периметр прямоугольника со сторонами A и B.
3. Занести в переменную P по очереди значения некоторых вышеприведенных функций, аргументом которых является сумма $A+B$.
4. Напечатайте случайное число в пределах от A до B.
5. Напечатайте значения переменной P в формате: 3 позиции для целой части и 2 для дробной.

3. Создание и отладка элементарной программы. Печать исходного текста. Комментарии.

Цели: отрабатывать навыки написания программ в среде программирования QBASIC, навыки работы с функциональными клавишами среды QBASIC.

Краткий обзор теоретического материала. Оператор комментария. Набор, отладка и запуск программы в среде программирования QBASIC.

Примеры возможных заданий.

1. Известны количество жителей (A) в государстве и площадь его территории (S). Определите плотность населения в государстве.

2. Даны два целых числа. Найти их среднее арифметическое. Модернизируйте программу так, чтобы на экран было выведено среднее геометрическое этих чисел.

3. Составить программу обмена значениями двух переменных.

Условный оператор

1. Условный оператор. Полная и неполная формы оператора.

Цели: познакомить учащихся с полной и сокращенной формой условного оператора. Рассмотреть типы задач, для которых лучше использовать полную и для которых – неполную форму ветвления.

Краткий обзор теоретического материала. Блок-схема конструкции «Ветвление». Условный оператор. Служебные слова IF, THEN, ELSE. Работа полного условного оператора. Работа сокращенного условного оператора.

Примеры возможных заданий.

1. Дано вещественное число. Вывести на экран его абсолютную величину (условно принимая, что соответствующей стандартной функции нет). Полный условный оператор не использовать.

2. Определите значение переменной x после выполнения фрагмента программы:

F = 5 : D = 7

IF F >= D THEN X = F ELSE X = D

3. Даны три целых числа. Вывести на экран те из них, которые являются четными.

4. Известны два расстояния: одно в километрах, другое в футах (1 фут=0,45 м). Какое из расстояний меньше?

5. Составить программу вычисления значений функции

$$Y = \begin{cases} \bar{x}, & \text{если } x > 0, \\ x^2, & \text{если } x \leq 0. \end{cases}$$

2. Вложенные условные операторы. Логические условия.

Цели: познакомить учащихся с использованием вложенных условных операторов и сложных (логических) условий при программировании различных задач.

Краткий обзор теоретического материала. Вложенные условные операторы. Блок – схема вложенных условных операторов. Конструкция вложенного условного оператора. Служебные слова IF, THEN, ELSE, END IF. Логические связки AND, OR, NOT. Программирование задач с использованием вложенных условных операторов и сложных условий.

Примеры возможных заданий.

1. Даны три числа. Составить программу вывода на экран наибольшего из них

- a) Используя вложенные условные операторы;
- b) Используя последовательные условные операторы.

2. Выяснить, в какой координатной четверти расположен треугольник, образованный прямой, заданной уравнением $y=ax + b$ и осями координат.

3. Определите значение переменной x после выполнения фрагмента программы:

$A = 53 : B = 14 : C = 2$

IF $A \geq B$ THEN

 IF $A \geq C$ THEN

$X = A - B - C$

 ELSE

$X = C - A - B$

 END IF

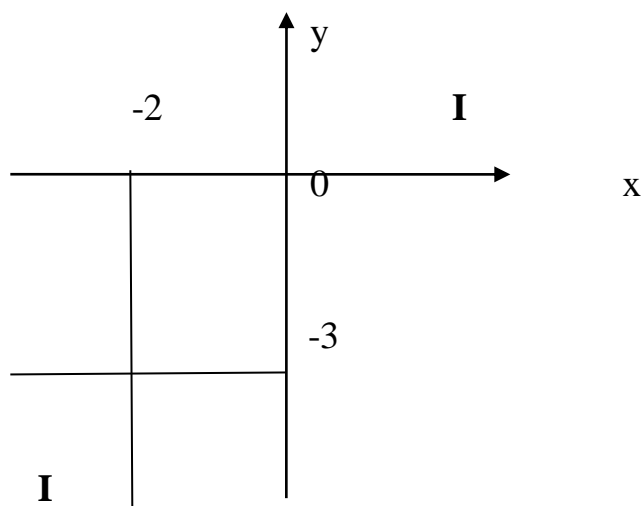
ELSE

$X = B - A$

END IF

4. Составьте программу для нахождения корней квадратного уравнения.

5. Определить, попадет ли точка с заданными координатами в область I (для простоты принять, что точка не попадает на границу этой области).



3. Оператор выбора.

Цели: познакомить учащихся с использованием оператора выбора (варианта) при программировании различных задач.

Краткий обзор теоретического материала. Блок-схема оператора варианта. Структура оператора варианта. Служебные слова SELECT, CASE, END SELECT.

Примеры возможных заданий.

1. С клавиатуры вводится какой-либо год XX века. Определить, мирным ли был этот год для России или она в то время участвовала в войне. Если год был военный, то выяснить, какая именно тогда была война.

2. Найти остаток от деления целого выражения $C=K(A+B)$ на 4 и вывести сообщение о величине остатка. Если остаток равен 0, то значение переменной C оставить без изменения, если 1 или 3 – уменьшить на величину остатка, если 2 – увеличить на величину остатка. Новое значение C вывести на экран (оно делится на 4 без остатка).

4. Программирование простых и условных вычислительных

алгоритмов. Вычисление простых и условных математических выражений.

Цели: запрограммировать различные задачи с использованием условного оператора.

Примеры возможных заданий.

1. Составить программу, которая определяет принадлежность точки с координатами (x, y) сектору, являющемуся пересечением круга радиуса R с центром в начале координат и I координатной четверти.

2. Составить программу, которая в зависимости от порядкового номера дня недели (1, 2, ..., 7) выводит на экран его название (понедельник, вторник, ..., воскресенье).

3. Составить программу для вычисления значения функции

$$Y = \begin{cases} -1, & \text{если } x < -1, \\ x, & \text{если } -1 \leq x < 1, \\ 1, & \text{если } x = 1. \end{cases}$$

Циклы

1. Оператор цикла с предусловием.

Цели: познакомить учащихся с циклом с предусловием, алгоритмом его работы; программировать задачи с использованием цикла с предусловием.

Краткий обзор теоретического материала. Блок-схема цикла с предусловием. Тело цикла. Оператор цикла с предусловием. Служебные слова WHILE, WEND. Работа оператора цикла с предусловием.

Примеры возможных заданий.

1. Дано натуральное число n . Посчитать количество цифр в числе.

2. Вычислить приближенное значение суммы $1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots$.

Суммирование продолжать до тех пор, пока очередное слагаемое не станет меньше заданного числа N .

3. Сумма в P рублей положена в сберегательный банк. При этом прирост составляет $M\%$ ежегодно. Считая, что прирост осуществляется непрерывно, написать программу, подсчитывающую, через какой промежуток времени сумма достигнет K рублей.

2. Оператор цикла с постусловием.

Цели: познакомить учащихся с циклом с постусловием, алгоритмом его работы; программировать задачи с использованием цикла с постусловием.

Краткий обзор теоретического материала. Блок-схема цикла с постусловием. Тело цикла. Оператор цикла с постусловием. Служебные слова DO, LOOP, WHILE, UNTIL. Работа оператора цикла с постусловием.

Примеры возможных заданий.

1. Вычислить приближенное значение суммы $\frac{1}{1*2} + \frac{1}{2*3} + \frac{1}{3*4} + \dots$.

Суммирование продолжать, пока очередное слагаемое (кроме первого) будет больше заданной точности вычислений.

2. Составить программу планирования закупки товара в магазине на сумму, не превышающую заданную величину.

3. Написать программу нахождения НОД двух натуральных чисел.

3. Оператор цикла с неизвестным числом повторений (параметром).

Цели: познакомить учащихся с циклом с параметром, алгоритмом его работы; программировать задачи с использованием цикла с параметром.

Краткий обзор теоретического материала. Блок-схема цикла с параметром. Тело цикла. Параметр цикла. Начальное и конечное значение параметра цикла. Шаг

цикла. Оператор цикла с параметром. Служебные слова FOR, TO, NEXT. Работа оператора цикла с параметром.

Примеры возможных заданий.

1. Вычислить:

$$\sum_{k=1}^{20} \frac{k+1}{k+2}$$

2. Составьте программу вычисления первых десяти членов последовательности, заданной формулой n-го члена: $X_n = 3^n - 5$.

3. Вычислить произведение всех двузначных нечетных чисел.

4. Определить, сколько из N заданных точек принадлежит графику $Y=|X|$.

4. Вложенность циклов. Программирование циклических алгоритмов.

Цели: дать понятие о вложенных циклах и особенностях их выполнения, научить программировать задачи с использованием вложенных циклов, задачи на сочетание цикла и условного оператора.

Краткий обзор теоретического материала. Конструкция вложенных циклов. Конструкция сочетания цикла и условного оператора.

Примеры возможных заданий.

1. Даны натуральные числа n и k. Составить программу вычисления выражения

$$1^k + 2^k + \dots + n^k.$$

2. *Старинная задача.* Сколько можно купить быков, коров и телят, если плата за быка 10 рублей, за корову – 5 рублей, за теленка – полтинник (0,5 рубля), если на 100 рублей надо купить 100 голов скота.

3. Написать программу, которая находит и выводит на печать все четырехзначные числа abcd, a, b, c, d – различные цифры, для которых выполняется : $ab - cd = a + b + c + d$.

4. Вывести на экран все целые числа от 100 до 200, кратные 3.

Массивы.

1. Одномерные массивы.

Цели: познакомить учащихся с понятием одномерного массива, рассмотреть способы задания одномерных массивов; учить программировать задачи с использованием одномерных массивов.

Краткий обзор теоретического материала. Понятие массива. Понятие индекса. Переменная с индексом. Простая переменная. Одномерные массивы. Описание массива:

DIM имя массива (n1 TO n2) AS тип элементов. Присвоение значений элементам массива. Способы задания одномерных массивов.

Примеры возможных заданий.

1. Опишите, как организовать вывод линейного массива в строку? В столбец?
2. Заполнить массив чисел так, чтобы
 - a. значения элементов совпадали с их индексами;
 - b. значениями элементов оказались случайные числа на интервале [111, 555];
 - c. каждый элемент вычислялся по формуле $b(i) = \frac{\sin i}{1 - |\sin i|}$
3. Сформировать одномерный массив из N элементов, элементы которого целые случайные числа в пределах от 1 до 45. Напечатать элементы массива в прямом и обратном порядке.
4. Дана последовательность $X_0=6765$, X_1 – неизвестно, $X_k = X_{k-1} + X_{k-2}$ ($k = 2, 3, \dots, 20$). Найти последовательность $X_0, X_1, X_2, \dots, X_{20}$, если известно, что $X_{20} = 0$.

Двумерные массивы.

Цели: познакомить учащихся с понятием двумерного массива, рассмотреть способы задания двумерных массивов; учить программировать задачи с использованием двумерных массивов.

Краткий обзор теоретического материала. Понятие матрицы. Двумерные массивы. Нумерация элементов двумерного массива. Способы описания двумерного массива. Способы задания двумерных массивов.

Примеры возможных заданий.

1. Определить, является ли данный квадратный массив симметричным относительно своей главной диагонали.
2. Заполнить массив A размером $m \times n$ в виде «змейки» (Например, если $n=5$, $m=5$, то массив должен быть следующим:

| | | | | |
|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 |
| 10 | 9 | 8 | 7 | 6 |
| 11 | 12 | 13 | 14 | 15 |
| 20 | 19 | 18 | 17 | 16 |
| 21 | 22 | 23 | 24 | 25 |

).
3. Составить программу, запрашивающую координаты ферзя на шахматной доске и показывающую поля доски, находящиеся под боем.

2. Поиск экстремальных значений величин в одномерных и двумерных массивах чисел.

Цели: программировать задачи на поиск экстремальных значений величин в одномерных и двумерных массивах чисел.

Краткий обзор теоретического материала. Проверка на четность. Подсчет количества элементов, отвечающих заданным условиям. Нахождение суммы,

произведения и количества элементов массива, отвечающих заданным условиям. Максимальный и минимальный элементы.

Примеры возможных заданий.

1. Найти максимальный (минимальный) элемент одномерного (двумерного) массива, а также его порядковый номер.

2. Посчитать количество, (сумму, произведение) отрицательных и неотрицательных (четных, нечетных) элементов одномерного (двумерного) массива.

3. Найти количество элементов, больших некоторого заданного числа X в каждой строке двумерного массива (во всем массиве).

3. Перестановка элементов массива. Сортировка массива.

Цели: программировать задачи, на перестановку и сортировку элементов массива.

Краткий обзор теоретического материала. Сортировка элементов массива. Метод «пузырька». Оператор SWAP. Перестановка элементов массива.

Примеры возможных заданий.

1. Отсортировать одномерный массив в порядке возрастания (убывания).

2. Сформировать одномерный массив из N элементов, элементы которого целые случайные числа в пределах от 1 до 45. Переставить элементы, стоящие на нечетных местах, с соответствующими элементами на четных местах.

3. Дан двумерный массив.

a. Поменять местами первую и третью строки.

b. Поменять местами второй и последний столбцы.

4. Слияние и отбор данных в одномерных и двумерных массивах.

Цели: программировать задачи на слияние и отбор данных (по определенному условию) в одномерных и двумерных массивах чисел.

Краткий обзор теоретического материала. Слияние и отбор данных в одномерных и двумерных массивах.

Примеры возможных заданий.

1. Произвести удаление (вставку) элемента одномерного массива, заданного датчиком случайных чисел.

2. Вставить строку из нулей на место строки с номером k двумерного массива.

3. Удалить строку с номером k двумерного массива.

4. В массиве $A(N, M)$ удалить все строки, сумма значений элементов которых отрицательна, и вставить перед последней строкой строку, значения элементов равны 1.

5. Поиск, подбор и группировка данных.

Цели: программировать задачи на поиск, подбор и группировку данных (по определенному условию) в одномерных и двумерных массивах чисел.

Краткий обзор теоретического материала. Поиск, подбор и группировка данных.
Примеры возможных заданий.

1. Дан массив $A(N)$, сформированный датчиком случайных чисел на интервале $(-50; 150)$. Из элементов массива A , кратных 5, сформировать массив B .
2. Дан двумерный массив размером 4×5 , заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен количеству отрицательных элементов, кратных 3 или 5, соответствующей строки.

Строковый, символьный тип данных

1. Строковый, символьный тип данных. Основные операции.

Цели: познакомить учащихся с основными сведениями о строковом, символьном типе данных, с операциями над символьными величинами.

Краткий обзор теоретического материала. Основные сведения о символьных величинах. Стандартные функции, процедуры для работы с символьными величинами: сравнение, конкатенация, копирование, удаление, замена (вставка), длина строки, подстрока числа и строки.

Примеры возможных заданий.

1. Распечатайте символы, соответствующие различным кодам, и одновременно определите, какой максимальный код можно использовать.
2. Вывести слово и:
 - a. Напечатать его по вертикали;
 - b. Переставить буквы слова в обратном порядке.
3. Составить программу «простейший справочник».
4. Даны две символьные строки A и B . Определите, в какой из них больше символов и на сколько.

2. Операции поиска и замены в символьных строках и массивах.

Цели: программировать задачи на поиск и замену элементов в символьных строках и массивах.

Краткий обзор теоретического материала. Удаление, замена (вставка), длина строки, подстрока числа и строки.

Примеры возможных заданий.

1. Определите, сколько раз в данном тексте встречается каждый символ (Один и тот же символ дважды не использовать).
2. Составить программу, заменяющую в данном тексте одну подстроку на другую.
3. Реализовать на компьютере возможность автоматического исправления слова <граматика> (правильно <грамматика>).

3. Программирование алгоритмов обработки текста.

Цели: программировать задачи на обработку, шифровку и дешифровку текста.

Краткий обзор теоретического материала. Понятие шифровки, дешифровки текста. Способы шифровки текста.

Примеры возможных заданий.

1. Дана строка, состоящая из нескольких слов, между словами стоит один пробел, в конце предложения - точка. Посчитать количество слов и вывести на экран только те из них, которые начинаются с буквы «А».
2. Составить программу устранения лишних пробелов в тексте (Лишним будем считать каждый пробел, следующий непосредственно за пробелом).
3. Определить, является ли данная фраза палиндромом
4. Заменить в тексте все символы их кодами.

Смотрите приложение.

Подпрограммы

1. Подпрограммы (функции и процедуры). Назначение. Способы описания.

Цели: дать учащимся представление о подпрограммах и возможностях их использования. Показать на примерах механизм реализации подпрограмм с помощью процедур и функций.

Краткий обзор теоретического материала. Понятие подпрограммы, процедуры, функции, Описание процедуры. Оператор SUB. Описание функции. Оператор FUNCTION. Механизм реализации подпрограмм с помощью процедур и функций. Обмен информацией между основной программой и подпрограммой. Глобальные и локальные переменные.

Примеры возможных заданий.

1. Составить процедуру сложения двух чисел, вводимых с клавиатуры.
2. Упорядочить значения трех переменных a, b, c используя подпрограмму упорядочивания двух переменных.
3. Написать функцию, подсчитывающую количество цифр целого числа. Используя ее, определить, в каком из двух чисел больше цифр.
4. Напишите функцию для нахождения НОД двух чисел с помощью алгоритма Евклида и используйте ее в программе для нахождения НОД уже N чисел.

2. Рекурсия.

Цели: дать учащимся представление о рекурсии и возможностях ее использования.

Краткий обзор теоретического материала. Понятие рекурсии. Рекурсивные алгоритмы. Задачи, сводимые к рекурсивным.

Примеры возможных заданий.

1. Перевести натуральное число из десятичной системы счисления в двоичную.
2. Определить, является ли заданное натуральное число простым.

3. Вычислить факториал натурального числа.
4. Найти НОД двух натуральных чисел.

Работа с файлами

1. Файлы. Текстовые файлы. Файлы с фиксированной структурой записи.

Цели: дать учащимся основные сведения о файлах и их классификации; познакомить с операциями над файлами; программировать задачи с использованием файлов.

Краткий обзор теоретического материала. Понятие файла. Классификация файлов. Операции над файлами: открытие файла, чтение и запись обрабатываемых данных, закрытие файлов.

Примеры возможных заданий.

1. Открыть файл f86.txt и сохранить в нем N целых чисел в пределах от 65 до 90:
 - а. Считайте информацию из данного файла;
 - б. Найдите сумму элементов, находящихся в файле f86.txt.
2. Создать файл, элементы которого вычисляются по формуле $M=I*I+4*I$; $I=1,2, \dots N$.
3. Создать текстовый файл, содержащий только целые числа, в каждой строке может быть несколько чисел, которые разделяются пробелами. Вывести на экран все числа с учетом разбиения на строки и подсчитать количество элементов в каждой строке.
4. Дан список данных о группе баскетболистов с указанием имени, роста, числа забитых мячей. Создайте файл, содержащий информацию о баскетболистах. Определите число записей в файле.

Повторение, решение задач

Цели: повторить основные разделы программирования; решать задачи.

Примечание. На данном уроке можно разобрать олимпиадные задачи по программированию.

Примеры возможных заданий.

1. Вася и Петя учатся в школе со следующим расписанием занятий: уроки начинаются в 8:30, каждый урок длится 45 минут, а между уроками есть пятнадцатиминутная перемена. Сейчас идет K-й урок по счету и Вася, уставший после напряженного учебного дня, поскорее хочет домой и спрашивает у Пети, сколько времени. Петя отвечает «Т минут», если с начала урока прошло Т минут и «минус Т минут», если до конца урока осталось Т минут. Составьте программу, которая по известному номеру урока, а также ответу Пети точно определит, сколько сейчас времени. (С клавиатуры вводятся два числа через пробел – K и T ($1 \leq K \leq 6$, $45 \leq T \leq 45$), где K – номер текущего урока, T – время, которое назвал Петя.)

2. Один из самых простых шифров – это шифр Цезаря. Его суть – скачек через две буквы алфавита в направлении от его начала к его концу: любая k -я буква, кроме последней и предпоследней, заменяется на $(k+2)$ -ю букву. Предпоследняя буква заменяется на первую, а последняя – на вторую букву.

Составить программу зашифровки и расшифровки русского текста шифром Цезаря.

3. Бесконечная в обе стороны последовательность чисел f_i , в которой любой целый элемент f_{i+2} вычисляется, используя следующее условие $f_{i+2}=f_{i+1}+f_i$.

Пусть заданы два различных члена этой последовательности f_i и f_j с соответствующими номерами i и j , а также некое целое число n . Необходимо восстановить элемент этой последовательности f_n , соответствующий номеру n .

4. Маленький мальчик взял лист бумаги $N \times N$ клеток и нарисовал на нем замкнутую m -звенную ломанную с вершинами в узлах клеток. После этого он выписал квадраты длин звеньев в порядке их обхода по ломанной и выкинул рисунок.

Определить, существует ли хотя бы одна ломанная, соответствующая данным, записанным мальчиком

$$1 \leq n \leq 15$$

$$3 \leq m \leq 20$$

Организация текущего и промежуточного контроля знаний. Творческие проекты (отчеты).

| № | Название |
|---|--|
| 1 | Творческий проект (отчет) по теме «История языков программирования». |
| 2 | Творческий проект (отчет) «Угадай пословицу». Цель. С помощью известных алгоритмических конструкций и изученных операторов языка Basic закодировать русские народные пословицы. |
| 3 | Творческий проект (отчет) одной задачи. Например, творческий проект (отчет) «Нахождение наибольшей из трех величин». Цель. Привести несколько способов решения одной задачи (используя последовательные условные операторы, используя логические условные операторы, используя оператор безусловного перехода и др.) |

Практические работы.

| № | Название |
|---|---|
| 1 | Практическая работа №1 «Составление простейших программ». |
| 2 | Практическая работа №2 «Ветвление». |
| 3 | Практическая работа №3 «Циклы». |

| | |
|---|---|
| 4 | Практическая работа №4 «Массивы». |
| 5 | Практическая работа №5 «Обработка текста» |
| 6 | Практическая работа №6 «Подпрограммы» |

Рекомендации для проведения практической работы.

Каждая практическая работа состоит из двух частей: теоретической и практической.

Теоретическая часть включает в себя пример разобранной задачи в изучаемом языке программирования и вопросы по данной теме, на которые ученик должен ответить, тем самым повторив основные моменты темы.

При выполнении практической части учащимся предлагается решить одну или несколько (по желанию) разноуровневых задач. Уровень 1 соответствует оценке «3», уровень 2 соответствует оценке «4», уровень 3 соответствует оценке «5».

Приложение.

Практическая работа №1.

Составление простейших программ.

Цель. Программировать задачи с использованием операторов ввода, вывода и присваивания.

1. Пример. Составить программу вычисления площади круга по известному радиусу.

QBASIC

```
INPUT «Ведите радиус круга R»; R
S=3.14*R^2
PRINT «Площадь круга равна»; S
```

PASCAL

```
program pl;
const k=3.14;
var r,s:real;
begin
  readln(r);
  s:=k*sqr(r);
  writeln(s);
end.
```

2. Вопросы.

1. Как оформляется оператор вывода на экран?
2. Что можно указать в качестве элементов списка вывода? Какой символ используется для разделения элементов списка вывода?
3. Что будет выведено на экран, если в списке вывода записано:
 - Число;
 - Имя величины;
 - Текст в кавычках;
 - Арифметическое выражение?
4. Как оформляется оператор ввода?
5. Что можно указать в качестве элементов списка ввода?
6. Как работает оператор ввода (что происходит при его выполнении)?
7. Как оформляется оператор присваивания? Как он работает (что происходит при его выполнении)?
8. Указать значения величин A и B после выполнения следующих операторов присваивания:
 - A=0
 - B=-9.99
 - B=A
 - A=b
9. Записать по правилам изучаемого языка программирования:
 - $5\cos y$
 - $-5 \bar{x} + \bar{y}$
 - $\sin\alpha\cos\beta + \cos\alpha\sin\beta$

- $mv^2 + e^x$

3. Выполните индивидуальное задание.

Индивидуальные задания.

Уровень 1.

1. Длины сторон прямоугольника равны А и В. Найдите его площадь, периметр, длину диагонали.
2. Градусная мера угла равна G, найдите его радианную меру R.
3. Вычислить и напечатать дискриминант квадратного уравнения.
4. Найти площадь кольца, если его толщина T - см, а диаметр внутреннего круга – R см.
5. Даны радиус основания цилиндра и высота цилиндра. Написать программу вычисления площади боковой поверхности и площади полной поверхности цилиндра.

Уровень 2.

1. По известному периметру равностороннего треугольника определить его площадь.
2. По известным диагоналям ромба вычислить и напечатать его периметр.
3. Даны периметр основания правильной пирамиды, и высота ее боковой грани. Написать программу вычисления площади боковой поверхности правильной пирамиды.
4. Вычислить значение выражения, предварительно задав значения x, y. $(\sin x)^3 y^2 \sqrt{2x^2 y - 2}$.
5. Вычислить значение выражения, предварительно задав значения переменных:

Уровень 3.

1. Дано расстояние в сантиметрах. Найти число полных метров в нем.
2. Дана масса в килограммах. Найти число полных центнеров в нем.
3. Даны два числа. Найти их сумму, разность, произведение, а также частное от деления первого числа на второе.
4. Известны координаты двух точек на плоскости. Составить программу вычисления расстояния между ними.
5. Составить программу обмена значениями двух переменных величин.
6. Составить программу обмена значениями трех переменных величин a, b, c по следующей схеме:
 - В присвоить значение c, a присвоить значение b, c присвоить значение a.

Практическая работа №2.

Ветвление.

Цель. Программировать задачи с использованием операторов условного и безусловного перехода.

1. Пример. Составить программу вычисления значения функции по формуле

$$Y = \begin{cases} \bar{x}, & \text{если } x \geq 0 \\ x^2, & \text{если } x < 0 \end{cases}$$

QBASIC

```
INPUT "Введите численное значение X"; X
IF X>=0 THEN Y=SQR(X) ELSE Y=X^2
PRINT "Значение функции равно";Y
```

PASCAL

```
program fn;
var x,y: real;
begin
  readln(x);
  if x>0 then begin
    y:=sqrt(x);
    write(y);
  end;
  else begin
    y:=sqr(x);
    write(y);
  end;
end.
```

2.Вопросы.

- Какие виды операторов условного перехода вы знаете?
- В каких случаях в программе используется полный оператор условного перехода? Как он оформляется? Как он работает (что происходит при его выполнении)? Нарисовать графическую схему выполнения.
- В каких случаях используют неполный оператор условного перехода? Как он оформляется? Как он работает (что происходит при его выполнении)? Нарисовать графическую схему выполнения.
- В каких случаях в программе используется вложенный условный оператор? Как он оформляется?
- В каких случаях в программе используется оператор варианта (выбора)? Как он работает (что происходит при его выполнении)? Нарисовать графическую схему выполнения.
- Исправить ошибки.

QBASIC

```
INPUT A,B,C
D:=A+B+C
IF D>0 THEN PRINT  $\sqrt{D}$  : GOTO 50
PRINT "√D=";D
```

PASCAL

```
program fn;
begin
  d=a+b+c
  if d>0 then
    writeln(√d);
  else
    write("d=");d
end.
```

3. Выполните индивидуальное задание.

Индивидуальные задания.

Уровень 1.

1. Увеличить число X на 10, если оно положительное, во всех остальных случаях уменьшить его на 10.

2. Написать программу, определяющую, является ли произвольное целое число X четным.

Написать программу, вычисляющую значение функции $Y = \frac{1}{1-a+b}$ для произвольных значений аргументов.

3. Написать программу, определяющую по длинам сторон A, B и C , является ли треугольник равнобедренным.

4. Группу детей, приехавших в лагерь отдыха, распределяют по отрядам в соответствии с возрастом:

С 6 до 7 лет – 1 отряд;

С 7 до 9 лет – 2 отряд;

С 9 до 11 лет – 3 отряд;

С 11 до 13 лет - 4 отряд;

С 13 до 15 лет – 5 отряд.

Написать программу, которая позволяет каждому приехавшему самому определить свой отряд. Продумать сообщение в случае, когда приехавшие по возрасту не подходят ни в один из отрядов: малыши (< 6 лет) и взрослые (> 15 лет).

5. Решить квадратное уравнение $ax^2+bx+c=0$.

Уровень 2.

1. Даны радиус круга и сторона квадрата. У какой фигуры площадь больше?

2. Даны два различных вещественных числа A и B . Написать программу возведения наименьшего из чисел в квадрат, наибольшего - в куб и нахождения среднего арифметического полученных значений.

3. Для произвольного числа A написать программу, возводящую его в квадрат, если число отрицательное; извлекающую из числа A квадратный корень, если положительное; оставляющую A без изменения, если $A=0$.

4. Фирма осуществляет набор сотрудников. Условия приема требуют не менее 3 лет рабочего стажа и возраст не более 35 лет. Написать программу, проверяющую, удовлетворяет ли кандидат этим требованиям.

5. Написать программу, выводящую на экран компьютера сообщение «Да», если точка X принадлежит отрезку $[A; B]$, и «Нет» в противном случае.

Уровень 3.

1. Составить программу для вычисления значения функции

$$Y = \begin{cases} -1, & \text{если } x < -1, \\ x, & \text{если } -1 < x < 1, \\ 1, & \text{если } x = 1. \end{cases}$$

3. Написать программу нахождения максимального из трех величин.
4. Составить программу нахождения суммы двух наибольших из трех различных чисел.
5. Составить программу нахождения произведения двух наименьших из трех различных чисел.

Практическая работа №3.

Циклы.

Цели. Программировать задачи с использованием повторений (циклов).

1.Пример. Составить программу вычисления суммы натуральных четных чисел от 2 до 30.

QBASIC

```
S=0
FOR I=2 TO 30 STEP 2
S=S+I
NEXT I
PRINT S
```

PASCAL

```
program summa;
  var s,i: integer;
  begin
    s:=0;
    for i:=2 to 30 step 2 do
      s:=s+i;
      writeln('s=',s);
  end.
```

2.Вопросы.

- В каких случаях используется оператор цикла с параметром? Как он оформляется? Как он работает (что происходит при его выполнении)? Нарисовать графическую схему выполнения.
- Что такое тело цикла?
- Может ли тело цикла с параметром не выполняться ни разу?
- Как должен быть оформлен оператор цикла с параметром, чтобы тело цикла выполнялось при уменьшающихся значениях параметра цикла? Как он будет работать (что будет происходить при его выполнении)? Нарисовать графическую схему выполнения.
- Можно ли в теле цикла с параметром не использовать величину – параметр цикла?
- В программировании существует правило: нельзя изменять параметр цикла в теле цикла. Приведите примеры, когда изменение параметра цикла приводит к его бесконечному выполнению.

- В каких случаях используются операторы цикла с условием?
- В каких случаях используется оператор цикла с предусловием? Как он оформляется? Как он работает (что происходит при его выполнении)? Нарисовать графическую схему выполнения.
- Может ли тело оператора цикла с предусловием:
А) не выполняться ни разу;
Б) выполняться бесконечное число раз (или до тех пор, когда пользователь прервет его выполнение)?
- В каких случаях используется оператор цикла с постусловием? Как он оформляется? Как он работает (что происходит при его выполнении)? Нарисовать графическую схему выполнения.
- Может ли тело оператора цикла с постусловием:
А) не выполняться ни разу;
Б) выполняться бесконечное число раз (или до тех пор, когда пользователь прервет его выполнение)?
- Всегда ли можно вместо оператора цикла с параметром использовать оператор цикла с предусловием? А наоборот?
- Всегда ли можно вместо оператора цикла с параметром использовать оператор цикла с постусловием? А наоборот?
- В программе написан бесконечный цикл с условием (с предусловием или постусловием).

Индивидуальные задания.

Циклы.

1 уровень.

1. Составить программу для вычисления суммы квадратов первых 100 членов натурального ряда.
2. Население города N увеличивается на 3% ежегодно. В текущем году оно составляет 40000 человек. Вывести предполагаемую численность населения города в каждый из ближайших 5 лет.
3. Каждый будний день недели рыбак ходил ловить рыбу. Сколько он поймал в среднем рыб, если известен ежедневный улов?
4. Напечатать ряд из повторяющихся чисел 20 в виде:
2020202020202020202020202020.
5. Напечатать «столбиком» все целые числа от 20 до 35.

Уровень 2.

1. Написать программу, которая распечатывает любой из столбиков таблицы умножения.
2. Каждая бактерия делится на две в течение 1 минуты. В начальный момент имеется 1 бактерия. Сколько их станет через 10 минут?
3. Одна штука некоторого товара стоит 20,4 руб. Напечатать таблицу стоимости 2, 3, ..., 20 штук этого товара.

- 4 Считая, что Земля - идеальная сфера с радиусом 6350 км, определить расстояние до линии горизонта от точки с высотой над Землей, равной 1, 2, ..., 10км.
- 5 Вывести «столбиком» следующие числа: 2,1, 2,2, 2,3, ..., 2,8.

Уровень 3.

1. В течение двух недель человек заключает пари на скачках. В первый день он делает ставку в 1 фунт и проигрывает. Во второй день, чтобы возместить свой проигрыш, он делает ставку в 2 фунта, однако снова проигрывает. На третий день он ставит 3 фунта, но опять проигрывает. Он упрямо продолжает увеличивать на 1 фунт свою ставку, но продолжает проигрывать. Написать программу, выводящую на экран сумму денег, проигранную этим человеком к концу второй недели.
2. Приблизительно можно считать, что скорость охлаждения горячего чая, налитого в стакан, пропорциональна разности между его температурой и температурой воздуха – за 1 минуту температура снижается на 3% от разности этих температур. Составить программу, реализующую эти вычисления и выводящую на экран температуру чая по истечении каждого из N промежутков времени по 1 минуте каждый. Начальная температура воздуха – 20⁰С, начальная температура чая – 100⁰С.
3. Резервуар наполнен 100л водного раствора, содержащего 5 кг растворенного сахара. Приток воды в сосуд составляет 6 л в минуту, а расход смеси из сосуда – 5л в минуту. Концентрация поддерживается равномерной по средствам помешивания. Написать программу, определяющую количество сахара, которое будет содержаться в резервуаре через 10 минут.
4. В некотором году (назовем его условно первым) на участке в 100 гектаров средняя урожайность ячменя составила 20 центнеров с гектара. После этого каждый год площадь участка увеличивалась на 5%, а средняя урожайность - на 2%.
Определить:
- урожайность за второй, третий, ..., восьмой год;
 - площадь участка в четвертый, пятый, ..., седьмой год;
 - какой урожай будет собран за первые шесть лет.
5. Определить суммарный объем в литрах 12 вложенных друг в друга шаров со стенками толщиной 5 мм. Внутренний диаметр внутреннего шара равен 10 см. Принять, что шары вкладываются друг в друга без зазоров.

Практическая работа №4.

Массивы.

Цели. Программировать задачи с использованием одномерных и двумерных массивов.

1.Пример 1. Найти сумму элементов одномерного массива.

QBASIC

```
INPUT N,K  
DIM A(N)
```

PASCAL

```
program summa;  
var c:=array [1:1000] of integer;
```

```
FOR I=1 TO N
A(I)=INT(2*K*RND(1) - K)
PRINT A(I);
NEXT I
S=0
FOR I=1 TO N
S=S+A(I)
NEXT I
PRINT "Сумма элементов ="; S
```

```
i,n : integer;
s : integer;
begin
write('Введите размер массива');
readln(n);
for i:=1 to n do
c[i]:=random(100);
s:=0
for i:=1 to n do
s:=s+c[i];
writeln(' ',s);
repeat until keypressed;
end.
```

Пример 2. Найти минимальный элемент двумерного массива.

QBASIC

```
INPUT N,K,M
DIM A(N,M)
FOR I=1 TO N
FOR J=1 TO M
A(I,J)=INT(2*K*RND(1)-K)
PRINT A(I,J);
массиве?');
NEXT J
PRINT
массиве?');
NEXT I
MIN=A(1,1)
FOR I=1 TO N
FOR J=1 TO M
IF A(I,J)<MIN THEN MIN=A(I,J)
NEXT J
NEXT I
PRINT "Минимальный элемент ="; MIN
```

PASCAL

```
program minimum;
var a:array [1:50;1:50] of integer;
i, j, n1, n2: integer;
min: integer;
begin
write('Сколько строк в
read(n1);
write('Сколько столбцов в
read(n2);
randomize;
for i:=1 to n1 do
for j:=1 to n2 do
begin
a[i,j]:=random(100);
clrscr;
end;
min:=a[1,1];
for i:=1 to n1 do
for j:=1 to n2 do
if a[i,j]<min then
min:=a[i,j];
```



```
writeln;  
writeln ('Минимальный элемент =');  
repeat until keypressed;
```

end.

2. Вопросы.

- Что такое одномерный массив? Для чего используются одномерные массивы? Как они описываются?
- Как в программе использовать значение конкретного элемента одномерного массива?
- Как называется номер элемента одномерного массива?
- Как можно заполнить одномерный массив?
- Для чего в программах используются двумерные массивы? Как они описываются?
- Сколько индексов характеризуют конкретный элемент двумерного массива?
- Как можно заполнить двумерный массив?
- Какую структуру данных описывает двумерный массив?
- Какой индекс двумерного массива изменяется быстрее при последовательном размещении массива в оперативной памяти?

3. Выполните индивидуальное задание.

Индивидуальные задания.

Уровень 1.

1. Найти все элементы массива целых чисел $A(20)$, меньшие 0.
2. Найти количество элементов массива целых чисел $C(40)$, больших квадрата первого элемента этого массива. Если таких элементов нет, выдать сообщение «поиск неудачен».
3. Найти и выдать на печать номера элементов массива целых чисел $B(k)$, которые больше первого, но меньше 4-го элементов массива. (Если таких элементов нет, выдать сообщение «таких элементов нет»).
4. Найти все нечетные элементы массива целых чисел $B(k)$.
5. Найти элементы массива целых чисел $C(k)$, которые делятся на 5 и на 8 (одновременно).

Уровень 2.

1. В линейном массиве целых чисел $B(25)$ заменить все элементы, меньшие 0, на квадрат этих элементов.
2. Заданы два массива. Создать один массив, в котором вначале идут четные элементы, затем нечетные.
3. В массиве хранится информация о количестве страниц в каждой из 100 книг. Все страницы имеют одинаковую толщину. Определить количество страниц в самой толстой книге.

4. В массиве хранится информация о стоимости 1 кг 20 видов конфет. Определить, сколько стоят самые дешевые конфеты.

5. В массиве хранится информация о росте 25 человек. Определить, на сколько рост самого высокого человека превышает рост самого низкого.

Уровень 3.

1. Составить программу, заменяющую значение любого элемента двумерного массива на число А.

2. Дан двумерный массив. Вывести на экран элемент, расположенный в правом верхнем и левом нижнем углу массива.

3. Заменить значения всех элементов главной диагонали массива на нулевые.

4. В двумерном массиве хранится информация о количестве студентов в той или иной группе каждого курса института с первого по пятый (в первой строке – информация о группах первого курса, во второй – второго и т.д.) На каждом курсе имеется 8 групп. Составить программу для расчета общего числа студентов на любом курсе.

5. Три группы студентов, в каждой из которых 20 человек, в сессию сдавали по 3 экзамена. Сведения об оценках каждой группы хранятся в двумерных массивах. Определить лучшую по средней оценке группу.

Практическая работа №5.

Символьные переменные.

Цель. Работать с текстом в языке программирования.

1.Пример 1. Определить, сколько раз в данном тексте встречается буква «А»

QBASIC

```
INPUT W$
S=0
FOR I=1 TO LEN(W$)
  IF MID$(W$,I,1)="A" THEN S=S+1
NEXT I
СИМВОЛОВ
PRINT S
```

PASCAL

```
program zadacha;
var s: string[20];
    i,k: integer;
begin
  writeln('Введите слово. Количество
  в нем не должно превышать 20');
  read(s);
  k:=0;
  for i:=1 to 20 do
    if s[i]='A' then k:=k+1;
  writeln ('буква А встречается', k, ' раз')
end.
```

Пример 2. В тексте заменить букву «А» на букву «Я».

QBASIC

```
INPUT W$
```

PASCAL

```
program zadacha;
```

| | |
|--|-------------------------|
| FOR I=1 TO LEN(W\$) | var s: string[20]; |
| | k: string[1]; |
| | g: string[1]; |
| IF MID\$(W\$,I,1)='А' THEN W\$=MID\$(W\$,1,I-1)+ | i:integer; |
| "Я"+MID\$(W\$,I+1,LEN(W\$)-I) | begin |
| NEXT I | writeln('Введите слово. |
| Количество | |
| PRINT "НОВОЕ СЛОВО"; W\$ | символов в нем не |
| должно | |
| превышать 20'); | |
| read(s); | |
| k:='А' | |
| g:='Я' | |
| for i:=1 to 20 do | |
| if s[i]=k then s[i]:=g | |
| for i:=1 to 20 do write(s[i]) | |
| end. | |

2. Вопросы.

- Что такое символьная (или строковая) переменная?
- Что собой представляет символьное выражение?
- Какие способы задания текста вы знаете?
- При помощи каких операторов можно задать значения символьных переменных?
- Какие операции определены для символьных переменных?
- Как определить длину строки?
- Какие стандартные функции определены для символьных переменных?
- Что напечатается в результате выполнения программы?

QBASIC

```
A$="ФУНКЦИЯ"  
L=LEN(A$)+4  
PRINT L
```

PASCAL

```
a:='функция';  
m:=length(a)+4;  
writeln(m);
```

3. Выполните индивидуальное задание.

Индивидуальные задания.

Обработка текста.

Уровень 1.

1. Составьте программу замены буквы «А» на букву «Б» и наоборот.

2. Используя функции символьных переменных, написать программу, которая из слова ИНФОРМАТИКА составит слово КИНО.
3. Используя функции символьных переменных, написать программу составления из слов ТЕРРИКОН, ОПЕРА, СЛАЙД слова ТРИОД.
4. Используя функции символьных переменных, определить, с какой позиции слово ФОРМА входит в слово ИНФОРМАЦИЯ.
5. Дано слово с ошибками: «малако». Исправить в нем ошибки, используя операцию склеивания, вывести на печать правильный вариант.

Уровень 2.

1. Составьте программу для замены в данном слове всюду части ДЕТЕКТИВ на часть РОМАН.
2. Составьте программу, проверяющую, является ли частью данного слова слово «СОК», ответ должен быть «да» или «нет».
3. Написать программу, которая выдает номер позиции заданной строки, в которой первый раз встречается буква Л. Если такой буквы нет, то выдать об этом сообщение.
4. Написать программу, определяющую длину самого короткого слова в заданной строке. Слова разделены одним пробелом.
5. Задана строка символов. Если ее длина превышает 20, то вывести первые 10 символов в одной строке, а все оставшиеся – в другой. Если длина строки не более 20, то вывести ее без изменения.

Уровень 3.

1. Составьте программу для определения, является ли данное слово «перевертышем».
2. Составьте программу для определения, является ли данная буква гласной, согласной или одной из букв Й, Ъ, Ь.
3. Задана строка, содержащая буквы и числа. Найти произведение всех чисел заданной строки.
4. Определить, сколько цифр содержится в записи произвольного вещественного числа.
5. Дан зашифрованный текст. Известно, что он зашифрован следующим образом: каждые четыре его символа записаны в обратном порядке. Если число символов в тексте не кратно четырем, то последняя группа символов не изменяется. Написать программу расшифровки текста.

Практическая работа №6.

Подпрограммы (процедуры, функции).

Цель. Программировать задачи с использованием подпрограмм, процедур и функций.

1. **Пример 1.** Составить процедуру нахождения максимального из двух действительных чисел, вводимых с клавиатуры. Используйте процедуру для нахождения максимального значения из четырех чисел.

QBASIC

```
DIM A, B, C, D, AS SINGLE
SUB maxim (X, Y, S)
real);
  IF X<Y THEN S=Y ELSE S=X
END SUB
```

CLS

```
INPUT "a=, b="; A, B
INPUT "c=, d="; C, D
CALL maxim (A, B, S)
CALL maxim (C, S, S)
CALL maxim (D, S, S)
PRINT "max="; S
  END
```

PASCAL

```
uses crt;
var a, b, s, c, d: real;
  procedure maxim (x, y: real; var s:
begin
  if x<y then s:=y else s:=x;
end;
begin
  clrscr
  writeln('a=b='); readln(a,b);
  writeln('c=d='); readln(c,d);
  maxim(a, b, s);
  maxim(c, s, s);
  maxim(d, s, s);
  writeln ('max=', s:3:1);
  readln;
end.
```

2. Вопросы.

- Что такое процедура?
- В чем различие между процедурой и функцией?
- Какие виды процедур и функций используются в программах?
- Перечислить несколько стандартных функций и процедур.
- В каких случаях целесообразно использовать нестандартные (определенные в программе) функции и процедуры?
 - Что такое формальные параметры функции или процедуры?
 - Как в программе выполнить функцию или процедуру?
 - Что такое фактические параметры функции или процедуры? Каковы правила их использования?
 - Какие преимущества дает использование нестандартных (определенных в программе) функций или процедур?
 - Что такое рекурсия?
 - Как оформляется рекурсивная процедура или функция?
 - Могут ли рекурсивные вызовы функции или процедуры продолжаться бесконечно? Как должна быть оформлена рекурсивная функция или процедура, чтобы количество рекурсивных вызовов было конечным?
 - Могут ли в одной программе процедура и функция иметь одинаковые имена?

- Может ли программист дать функции имя \sin ?

3. **Выполните индивидуальное задание.**

Индивидуальные задания.

Подпрограммы (процедуры, функции).

Уровень 2.

1. Рассчитать значение x , определив и использовав необходимую функцию. $x = \frac{6 + \sqrt{6}}{2} + \frac{13 + \sqrt{13}}{2} + \frac{21 + \sqrt{21}}{2}$

2. Определить значение $z = \max(a, 2b) * \max(2a-b, b)$, где $\max(x, y)$ есть максимальное из чисел x, y . При решении задачи не использовать функцию \max .

3. Составить программу, в результате которой величина A меняется значением c величиной B , а величина C – с величиной D . (Определить процедуру, осуществляющую обмен значениями двух переменных величин).

4. Даны основания и высоты двух равнобедренных трапеций. Найти сумму их периметров и сумму их площадей. (Определить процедуру для расчета периметра и площади равнобедренной трапеции по ее основаниям и высоте).

5. Даны первый член и знаменатель геометрической прогрессии. Написать рекурсивную функцию нахождения суммы n первых членов прогрессии.

Уровень 3.

1. Найти периметр фигуры $ABCD$ по заданным сторонам AB, AC, DC . (Определить функцию для расчета гипотенузы прямоугольного треугольника по его катетам. $\angle BAC = 90^\circ, \angle BCD = 90^\circ$).

2. Даны три квадратных уравнения

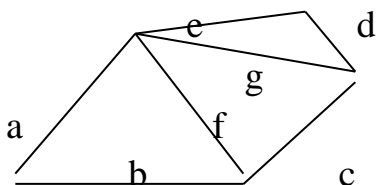
$$Ax^2 + Bx + C = 0$$

$$Bx^2 + Ax + C = 0$$

$$Cx^2 + Ax + B = 0.$$

Сколько из них имеют вещественные корни? (Определить функцию, позволяющую распознавать наличие вещественных корней в квадратном уравнении).

3. Даны вещественные числа a, b, c, d, e, f, g . Найти площадь пятиугольника, изображенного на рисунке. (определить функцию для расчета площади треугольника по трем его сторонам).



4. Даны два предложения. Найти общее количество букв «н» в них. (Определить функцию для расчета количества букв «н» в предложении).

5. *Написать рекурсивную процедуру перевода натурального числа из десятичной системы счисления в двоичную.