

АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОРГАНИЗАЦИЯ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ИНФОСФЕРА КОЗЬМОДЕМЬЯНСК»

Рассмотрена и принята
на заседании
педагогического совета,
протокол
от 23 июня 2026г. № 13

УТВЕРЖДЕНА
приказом АНО ДО
«Инфосфера Козьмодемьянск»
от 29 июня 2026г. № 29.06.1-од

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«ПРОГРАММИРОВАНИЕ. ТОЧКА ВХОДА 2.0»
(ПТв 2.0)

Направленность: техническая

Возраст обучающихся: 15-17 лет

Срок реализации программы: 9 мес. (сентябрь – май)

Объем программы: 102 академических часа

Разработали:

Романов А.В., Тарьянова А.Л.

Козьмодемьянск, 2026

Пояснительная записка

В настоящее время, наше общество находится на этапе глобальной информатизации и компьютеризации, поэтому возрастает потребность в специалистах с высоким уровнем владения информационными компетенциями, отвечающих социальному заказу по подготовке квалифицированных кадров в области программирования, а также обладающих высоким интересом к ИТ-сфере.

Одной из составляющих информационной компетентности является владение языком программирования. Встаёт вопрос о выборе языка программирования, который отвечает современным требованиям к написанию программ, служит основой для дальнейшего развития и совершенствования программистских компетенций.

Есть различные языки программирования, но одни могут отличаться от других по мере их использования. Дефицит порождает спрос, поэтому в данный момент идет спрос на языки с дальнейшей поддержкой различных платформ.

Для обучения программированию нами выбран классический язык программирования - Pascal.

Особенностями языка программирования Pascal являются строгая типизация и наличие средств структурного программирования. По мнению Вирта, язык должен способствовать дисциплинированному программированию, поэтому, наряду со строгой типизацией, в Pascal сведены к минимуму возможные синтаксические неоднозначности, а сам синтаксис автор постарался сделать интуитивно понятным даже при первом знакомстве с языком.

Одной из целей создания языка Pascal является обучение учащихся структурному программированию. До сих пор Паскаль заслуженно считается одним из лучших языков для начального обучения программированию. Его современные модификации, такие как Object Pascal, широко используются в промышленном программировании (среда Delphi). Также на основе синтаксиса языка Pascal создан язык программирования Structured text (ST) или Structured Control Language (SCL) для программируемых логических контроллеров.

Научившись программировать на языке Pascal, учащиеся получат мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность их конструкций позволит учащимся потом с лёгкостью выучить любой другой язык программирования.

Дополнительная общеобразовательная общеразвивающая программа «Программирование. Точка входа 2.0» (далее - программа) имеет техническую направленность и предназначена для обучающихся 15-17 лет средних общеобразовательных школ.

Целью общеразвивающей программы является систематизация школьных знаний по математике и информатике, а также изучение основ

программирования на примере языка PascalABC.NET и выполнение законченного проекта.

Задачи:

1. Формирование системного подхода к анализу окружающей действительности.
2. Формирование представлений об истории развития языков программирования.
3. Формирование представлений о базовых формах записи алгоритмов (словесная, блок-схема, алгоритмический язык, язык программирования высокого уровня).
4. Формирование навыков построения базовых конструкций алгоритмов: последовательного (линейного), циклического, разветвляющегося, вспомогательного на языках программирования высокого уровня. (Pascal ABC.NET)
5. Формирование навыков создания в среде Pascal ABC.NET программ для обработки информации различных видов: числовой, текстовой, графической и звуковой.
6. Развитие у обучающихся алгоритмического, логического и творческого мышления, пространственного воображения, словарного запаса технической направленности и интереса к инженерным специальностям.
7. Развитие коммуникативных способностей, формирование умения командной работы и навыка коллективного творчества.
8. Создание оптимальных условий для формирования информационной культуры школьников в рамках дополнительного образования.

Категория обучающихся: лица в возрасте 15-17 лет.

На обучение по программе принимаются все желающие.

Режим занятий и распределение учебного времени и времени отдыха приведены в Приложении № 1.

Форма обучения - очная. Формы организации учебной деятельности: коллективная, групповая, парная, индивидуальная. Разнообразие форм занятий неразрывно связано с содержанием, целями занятия, возрастными особенностями учащихся. Формы и виды занятий: лекция классическая, лекция проблемная, лекция-визуализация, лекция-диалог, аудиторно-практическое занятие классическое, практикум-лабораторная работа, самообучение, презентация, защита проекта.

На занятиях строго соблюдаются Санитарно-эпидемиологические нормы: учтены требования к технике, освещению, продолжительности занятий; проводятся профилактические упражнения для глаз и физкультминутки.

Объем, срок освоения программы. Программа рассчитана на 9 месяцев, 34 учебные недели, и построена на принципе постоянного усложнения и обогащения материала. Общий объем программы – 102 академических часа.

Планируемые результаты освоения программы

Личностные результаты

1. 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.

2. Навыки сотрудничества со сверстниками, взрослыми в образовательной, учебно-исследовательской, проектной и других видах деятельности.

3. Эстетическое отношение к миру, включая эстетику научного и технического творчества.

4. Осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.

5. Отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

Метапредметные результаты

1. Овладение способностью принимать и сохранять цели и задачи учебной деятельности, поиск средств ее осуществления:

- умение выполнить алгоритм, приводящий к решению задачи;
- умение сформулировать задачу, определить необходимые для решения данные, разделить их на имеющиеся и недостающие, провести поиск недостающих данных.

2. Формирование умения планировать, контролировать и оценивать учебные действия в соответствии с поставленной задачей и условиями ее реализации; определять наиболее эффективные способы достижения результата:

- освоение понятия «алгоритм»; выполнение алгоритмов;
- понимание наличия в любой системе противоречий;
- понимание диалектического единства противоположностей, перехода свойств из разряда «плохих» в разряд «хороших» и обратно в зависимости от ситуации;
- понимание невозможности абсолютного превосходства одной из альтернативных систем над всеми остальными;
- понимание обязательного наличия недостатков у любой системы, невозможности исправить все недостатки, необходимости «платить» за их исправление, умения оценить сравнительную значимость недостатков;
- умение представить информацию в наиболее удобном виде.

3. Формирование умения понимать причины успеха/неуспеха учебной деятельности и способности конструктивно действовать даже в ситуациях неуспеха:

- понимание наличия в любой системе противоречий;

- понимание диалектического единства противоположностей, перехода свойств из разряда «плохих» в разряд «хороших» и обратно в зависимости от ситуации; \понимание невозможности абсолютного превосходства одной из альтернативных систем над всеми остальными;
- понимание обязательного наличия недостатков у любой системы, невозможности исправить все недостатки, необходимости «платить» за их исправление, умения оценить сравнительную значимость недостатков.

4. Освоение начальных форм познавательной и личностной рефлексии.

5. Овладение логическими действиями сравнения, анализа, установления аналогий и причинно-следственных связей, построения рассуждений:

- изучение элементов классической логики (суждения, противоположные суждения, логические операции, таблицы истинности, использование таблиц решений, характеристических таблиц);
- изучение элементов диалектической логики (понятие противоречия);
- построение цепочек причинно-следственных связей;
- сравнение объектов друг с другом;
- проведение рассуждений, связанных с противоречиями.

6. Готовность слушать собеседника и вести диалог; готовность признавать возможность существования различных точек зрения и права каждого иметь свою; излагать свое мнение и аргументировать свою точку зрения и оценку событий.

Предметные результаты

1. Освоение принципов построения базовых математических структур, необходимые для истолкования компьютерных программ, базовых методов сравнения эффективности различных алгоритмов, основных методов доказательств корректности алгоритмов.

2. Овладение основами логического и алгоритмического мышления:

- изучение основ алгебры логики;
- изучение базовых логических операций, правил построения простых и сложных логических выражений;
- умение находить значение логических выражений с помощью таблиц истинности;
- умение строить и анализировать логические схемы, решать текстовые логические задачи.

3. Умение действовать в соответствии с алгоритмом и создавать алгоритмы различной структуры, работать с таблицами, схемами, графиками и диаграммами, цепочками, совокупностями, представлять, анализировать и интерпретировать данные:

- изучение алгоритмики (понятие алгоритма, способы записи алгоритмов, виды алгоритмов, умение исполнять алгоритмы);

- умение представлять совокупности взаимосвязанных данных в виде таблиц, схем, графиков, диаграмм;

- умение реализовывать поисковые алгоритмы, структурировать, анализировать полученную информацию и выбирать наиболее эффективные способы ее представления.

4. Развитие навыков программирования на языках высокого уровня:

- умение определить и описать в программе перечень и типы ресурсов компьютера, необходимых для решения поставленной задачи;

- умение изложить словесный алгоритм решения задачи;

- умение представить алгоритм на одном из языков программирования;

- навыки использования в структуре программного кода всех основных алгоритмических структур;

- навыки самостоятельного тестирования работоспособности алгоритма при разных входных данных;

- навыки определения эффективности алгоритма и выбора наиболее оптимального по ресурсам и времени алгоритма решения задачи.

5. Приобретение навыков структурного программирования, при котором разработка алгоритма происходит блочно, с выделением подзадач, описываемых с помощью вспомогательных алгоритмов.

Программа «Программирование. Погружение» является программой с разделами для изучения.

Содержание обучения по данной программе представлено следующими разделами:

1. Язык программирования PascalABC.NET
2. Основные алгоритмические структуры
3. Линейные программы
4. Ветвление
5. Циклы
6. Массивы
7. Строки
8. Процедуры и функции
9. Файлы. Процедуры и функции работы с файлами
10. Модуль GraphABC
11. Графические объекты. Модуль ABCObjects
12. Проект. Написание игрового приложения.

Теория в программе излагается достаточно кратко, поскольку основной уклон сделан на практические занятия. По этой причине основные идеи программы доступны слушателям без опыта программирования. Дополнительное изучение материала осуществляется за счет самостоятельного изучения материалов, предоставляемых преподавателем. Источниками материала для изучения являются методические пособия, техническая литература, профессиональные ресурсы Интернет.

Закрепление теоретического материала осуществляется при проведении лабораторных на компьютерах, выполнения проблемно-

ориентированных, поисковых, творческих заданий, контрольных работ, написание и защита проекта.

Учебно-тематический план Программирование. Точка входа 2.0

№	Наименование раздела, темы	Всего часов	В том числе	
			Теория	Практика
1	Язык программирования PascalABC.NET	2	2	-
2	Основные алгоритмические структуры	2	1	1
3	Понятие переменной: имя, тип данных, значение	2	1	1
4	Линейные программы	6	1	5
4.1	Модуль CRT			
4.2	Линейные программы в PascalABC.NET			
4.3.	Линейные программы в сравнении с другими языками программирования			
5	Ветвление	11	2	9
5.1	Постое ветвление и логические условия. Решение типовых задач в PascalABC.NET			
5.2	Постое ветвление и логические условия. Решение задач разного уровня сложности			
5.3	Неполный и полный оператор ветвления. Решение задач			
6	Контрольная работа «Ветвление и логические условия»	1	1	-
7	Циклы	12	3	9
7.1	Цикл WHILE. Решение задач.			
7.2	Цикл Repeat until. Решение задач разного уровня сложности			
7.3	Цикл FOR. Решение задач.			
8	Массивы	9	2	7
8.1	Одномерные массивы в PascalABC.NET			
8.2	Двумерные массивы в PascalABC.NET			
8.3.	Творческое задание «Новогодняя открытка»	4	-	4
9	Строки	6	1	5
10	Контрольная работа «Циклы и массивы»	1	1	-
11	Процедуры и функции	9	2	7
12	Файлы. Процедуры и функции работы с файлами	6	1	5
13.	Модуль GraphABC	12	3	9
13.1	Графические примитивы модуля GraphABC. Решение задач.			
13.2	Движение фигур			
13.3	Управление движением фигур с клавиатуры			

13.4	Игра «Крестики-нолики». Массивы. Обработка событий, связанных с мышкой.			
14	Графические объекты. Модуль ABCObjects	9	3	6
15	Проект. Написание игрового приложения	10	2	8
15.1	Этапы работы над проектом. Старт игры			
15.2	Обработка событий игры			
15.3	Окончание игры. Тестирование и отладка игры			
15.4	Защита проекта			
Итого		102	26	76

Содержание

Тема 1. Язык программирования PascalABC.NET

Структура программы на языке Паскаль. Рекомендации по стилю записи программы, использование комментариев. Алфавит языка. Организация ввода-вывода. Оператор присваивания.

Тема 2. Основные алгоритмические структуры

Арифметические выражения. Стандартные функции. Правила записи арифметических выражений. Операции. Операнды. Следование. Условный оператор. Оператор выбора. Организация циклических структур.

Тема 3. Понятие переменной: имя, тип данных, значение

Типы данных: целый и вещественный, логический и символьный. Константы. Переменные. Именованые переменных.

Тема 4. Линейные программы

Линейные программы. Основные математические операции. Задачи на вывод данных. Модуль CRT.

Тема 5. Ветвление

Простое ветвление и логические условия. Неполный и полный оператор ветвления. Вложенное ветвление. Решение задач разного уровня сложности.

Тема 6. Контрольная работа «Ветвление и логические условия»

Выполнение теста по пройденному материалу

Тема 7. Циклы

Цикл WHILE. Цикл REPEAT UNTIL. Цикл FOR. Решение задач разного уровня сложности.

Тема 8. Массивы

Объявление и основные операции над массивами. Максимальный и минимальный элемент массива. Поиск в массиве элементов с заданными свойствами. Перебор элементов массива.

Двумерные массивы. Действия над элементами массива. Обработка элементов двумерных массивов. Квадратная матрица. Транспонирование матрицы.

Тема 9. Строки

Строковый тип данных. Нахождение, замена, вывод на экран элементов строк, подчиненных определенным условиям с использованием стандартных функций.

Тема 10. Контрольная работа «Циклы и массивы»

Выполнение теста по пройденному материалу.

Тема 11. Процедуры и функции

Процедуры. Функции. Рекурсии. Процедуры и функции пользователя.

Тема 12. Файлы. Процедуры и функции работы с файлами

Работа с файлами. Обработка текстовой информации.

Тема 13. Модуль GraphABC

Операции с графическим окном. Встроенные функции для работы с графическим окном. Графические примитивы модуля GraphABC. Решение задач. Движение фигур. Написание программы случайного блуждания частицы в поле.

Тема 14. Графические объекты. Модуль ABCObjects

Модули ABCObjects. Класс: конструктор, свойства, методы. Стандартные графические объекты. Работа с графическими файлами. Классы PictureABC и MultipictureABC. Движение объектов. События модуля GraphABC. Работа с клавиатурой. События модуля GraphABC. Работа с мышкой.

Тема 15. Проект. Написание игрового приложения.

Этапы работы над проектом. Старт игры. Уровни сложности. Обработка событий игры. Окончание игры. Тестирование и отладка игры. Защита проекта.

Формы аттестации обучающихся.

Промежуточная аттестация обучающихся проводится дважды за период обучения по программе в форме контрольных работ с целью повышения ответственности педагогов и обучающихся за результаты образовательного процесса, за степень усвоения обучающимися программы в рамках учебного года.

Итоговая аттестация по данной программе проводится в конце обучения для определения достижения планируемых результатов обучающимися в форме написания и защиты проекта.

По итогам обучения по программе выдается свидетельство (см. Приложение №3).

Организационно-педагогические условия реализации программы

Реализация программы обеспечена следующим оборудованием: столы; стулья; рабочие компьютеры; робототехнические конструкторы; компьютер педагога; мультимедийный проектор; экран для проектора; магнитная доска для учебной аудитории; выход в Интернет.

Основное программное обеспечение:

Система программирования PascalABC.NET

Свободно распространяемая версия PascalABC.NET MiniPack

Условия для занятий соответствуют санитарно-гигиеническим нормам.

Наполняемость учебной группы 10 человек.

Учебно-методическое и информационное обеспечение программы

1. Бешенков С. А., Ракитина Е. А. Информатика в школе: Приложение к журналу «Информатика и образование». №1 - 2005. Москва: Образование и Информатика, 2005.
2. Бешенков С. А., Ракитина Е. А. Моделирование и формализация. Методическое пособие. Москва: Лаборатория Базовых Знаний, 2002.
3. Копыльцов А. В. Компьютерное моделирование: сферы и границы применения. Методическое пособие. Санкт-Петербург.: СММО Пресс, 2005.
4. Семакин И. Г. Информационные системы и модели. Элективный курс: Учебное пособие. Москва: Бином. Лаборатория знаний, 2005.
5. Суворов Н. И. Информационное моделирование. Величины, объекты, алгоритмы. Москва: Лаборатория Базовых Знаний, 2002.
6. Угринович Н. Д. Информатика и ИКТ. Москва: Бином. Лаборатория знаний, 2005.
7. Фридланд А. Я. Информатика: процессы, системы, ресурсы. Москва: Бином. Лаборатория знаний, 2003.
8. [Электронный ресурс] <http://pascalabc.net/> - официальный сайт разработчиков среды программирования PascalABC.NET.
9. [Электронный ресурс] <http://ds-release.ru/video-uroki-po-pascalabc-net/> - видеоуроки по PascalABC.NET
10. [Электронный ресурс] - http://elar.urfu.ru/bitstream/10995/28702/1/978-5-7996-1260-3_2014.pdf - Л.И. Долинер. Основы программирования в среде PascalABC.NET. Учебное пособие.
11. [Электронный ресурс] - <http://informatics.mccme.ru/login/index.php> - дистанционная подготовка по программированию.
12. [Электронный ресурс] - <https://myklad.org/5/2/7/zanimatelnye-uroki-s-paskalem-ili-pascalabc-net-dlya-nachinayushhix-rubancev-valerij-2013-programmirovanie-pdf-otlichnyj-skan-s-ocr.html> - В. Рубанцев. Занимательные уроки с Pascal. Учебное пособие.
13. [Электронный ресурс] - <http://kpolyakov.spb.ru/> - сайт автора учебников по информатике и программированию К. Полякова.

Планируемые сроки и режим занятий
(в соответствии с порядковым номером учебной недели)

№ п/п	Дата / неделя	Число часов	№ п/п	Дата / неделя	Число часов
1	-	-	18	неделя	3
2	неделя	4	19	неделя	3
3	неделя	3	20	неделя	3
4	неделя	3	21	неделя	3
5	неделя	3	22	неделя	3
6	неделя	3	23	неделя	3
7	неделя	3	24	неделя	3
8	неделя	4	25	неделя	3
-	каникулы	-	26	неделя	4
9	неделя	3	-	каникулы	-
10	неделя	3	27	неделя	4
11	неделя	3	28	неделя	3
12	неделя	3	29	неделя	3
13	неделя	3	30	неделя	3
14	неделя	3	31	неделя	3
15	неделя	3	32	неделя	3
16	неделя	4	33	неделя	4
	каникулы				
17	неделя	3			

1. Учебных недель: 32 недели по 3-4 часа в неделю.
Режим занятий: один урок 45 минут, перерыв между занятиями 10-15 минут.
Общая продолжительность занятий в день у одной группы – не более 4-х академических часов с обязательными перерывами.
2. Промежуточная аттестация проводится в декабре, январе.
Итоговая аттестация обучающихся – май.
3. Сроки и продолжительность каникул:
Осенние каникулы – 7 дней
Зимние каникулы – 10 дней
Весенние каникулы – 7 дней
Летние каникулы – 3 летних месяца.
4. Праздничные выходные дни во время учебного года:
4 ноября – День народного единства,
1-8 января - Новогодние каникулы,
23 февраля - День защитника Отечества,
8 марта - Международный женский день,
1 мая - Праздник Весны и Труда,
9 мая - День Победы,

5. Окончание учебного года – май текущего года (зависит от реализации дополнительной общеобразовательной общеразвивающей программы).

Приложение № 2

**Демонстрационный вариант
Промежуточного тестирования по курсу программы**

Контрольная работа «Ветвление и логические условия»

1. В результате выполнения фрагмента программы, представленной ниже, переменная С примет значение: $a:=5$; $b:=20$; $a:=a-b/2$; if $a>b$ then $c:=a+b$ else $c:=b-a$;

а) 25

б) -15

в) 5

72

2. В результате выполнения представленной ниже программы, произойдет:

```
program pr;
```

```
var a, b, c: integer;
```

```
begin write ('Введите первое целое число');
```

```
readln (a);
```

```
write ('Введите второе целое число');
```

```
readln (b); if a3 then c:=4 else c:=5;
```

а) $a = 1$, $b = 4$

б) $a = 3$, $b = 3$

в) $a = 4$, $b = 1$

3. В результате выполнения фрагмента программы, представленной ниже, переменная С примет значение 6, если: if $a<3$ then if $b>3$ then $c:=4$ else $c:=5$ else $c:=6$;

а) $a = 4$, $b = 1$

б) $a = 1$, $b = 4$

в) $a = 5$, $b = 3$

4. Необходимо записать неполный условный оператор, обозначающий неполные ветвления на языке Паскаль:

а) if <условие> then

б) if <условие> then

в) then <условие> if

5. Конструкция вида begin <последовательность операторов> end называется:

а) отставной оператор

б) уставной оператор

в) составной оператор

6. В результате выполнения фрагмента программы, представленной ниже, переменная С примет значение:

a:=20; b:=(a mod 10)*10; a:=(b*2)+a div 100; if a>b then c:=a+b else c:=b-a;

а) 0

б) 1

в) 2

7. Что произойдет в результате выполнения программы, представленной ниже:

```
program pr var
```

```
a, b, c: integer;
```

```
begin write ('Введите первое целое число');
```

```
readln (a);
```

```
write ('Введите второе целое число'); readln (b); if a>b then c:=2*a else  
c:=2*b; writeln (c) end.
```

а) уменьшение большего из двух чисел вдвое

б) увеличение меньшего из двух чисел вдвое

в) увеличение большего из двух чисел вдвое

8. Условный оператор целесообразно использовать именно в этой программе:

а) подсчитывающей количество символов

б) сравнивающей значения переменных

в) вывода на печать значения константы

9. В результате выполнения фрагмента программы, представленной ниже, переменная С примет значение 4, если:

if a>3 then if b>3 then c:=4 else c:=5;

а) a=4, b=4

б) a=3, b=3

в) a=4, b=1

10. В результате выполнения фрагмента программы, представленной ниже, переменная С примет значение 4, если:

if a<4 then if b>3 then c:=4 else c:=5 else c:=6;

а) a = 4, b = 1

б) a = 1, b = 4

в) a = 5, b = 3

11. Необходимо записать условный оператор, обозначающий разветвляющиеся алгоритмы на языке Паскаль:

а) if <условие> then <оператор_1> else <оператор_2>

б) if <условие> then

в) then <условие> if

12. Необходимо записать название оператора, который применяется при необходимости выполнения определенной последовательности операторов при некотором условии:

- а) уставной оператор
- б) подставной оператор
- в) составной оператор

13. Перед else знак «;» :

- а) не ставится
- б) ставится
- в) ставится после

14. В качестве условий используется одно из таких логических выражений:

- а) дополнительное
- б) простое
- в) основное

15. В качестве условий используется одно из таких логических выражений:

- а) заключенное
- б) главное
- в) сложное

16. Условия, записанные с помощью операций отношения:

- а) главные
- б) простые
- в) сложные

17. Условия, записанные с помощью логических операций:

- а) основные
- б) простые
- в) сложные

18. В условном операторе и после then, и после else можно использовать:

- а) несколько операторов
- б) только один оператор
- в) только восемь операторов

19. В качестве оператора после then и else можно использовать:

- а) условный оператор
- б) дословный оператор
- в) естественный оператор

20. Если последовательность выполнения шагов алгоритма изменяется в зависимости от выполнения некоторых условий, то это такой алгоритм:

- а) линейный
- б) разветвляющийся
- в) циклический

21. В алгоритмах ветвления, или условного перехода, присутствует ... ветвь(ей) вычислительного процесса:

- а) ноль
- б) одна
- в) несколько

22. Для реализации разветвляющегося алгоритма в языке VBA есть:

- а) четыре оператора
- б) два оператора
- в) три оператора

23. Оператор выбора:

- а) Select Choice
- б) If
- в) Case

24. Для записи условий в операторе if используются ... выражения:

- а) логические
- б) условные
- в) главные

25. Логическое отрицание:

- а) No
- б) Not
- в) Now

Контрольная работа «Циклы и массивы»

1. Алгоритм, в котором команда или серия команд повторяется, называется

- а) циклическим;
- б) линейным;
- в) разветвляющимся

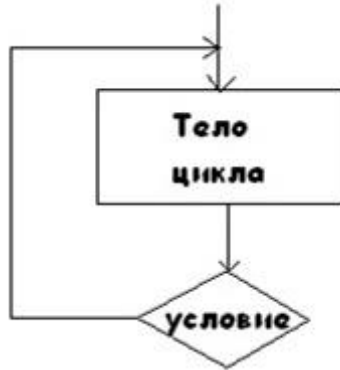
2. Укажите процесс, который демонстрирует работу циклического алгоритма:

- а) движение часовой стрелки;
- б) движение поезда от начальной до конечной остановки;
- в) направо пойдешь - коня потеряешь, налево пойдешь - ничего не найдешь.

3. Какой цикл называют циклом с предусловием?

- а) Repeat;
- б) While;
- в) For.

4. Какой цикл изображен на блок схеме?



- а) While;
- б) Repeat;
- в) For.

5. Параметр цикла For может получить значения...

- а) 1,2,3,4,5;
- б) 2,4,6,8,10;
- в) 1,3,5,7,9.

6. Требуется подсчитать сумму натуральных чисел от 5 до 125. Какое условие нужно использовать в цикле While?

- а) $i > 125$;
- б) $i < 125$;
- в) $i \leq 125$.

7. Сколько раз будет выполняться цикл For $i:=7$ to 12 do...;

- а) 5 раз;
- б) 6 раз;
- в) 4 раза.

8. Сколько раз исполнится цикл:

$i:=4$;
while $i < 10$ do $i:=i+3$;

- а) 2 раза;
- б) 3 раза;
- в) 4 раза.

9. Сколько раз исполнится следующий цикл:

```
i:=21;
```

```
Repeat
```

```
i:=i-5
```

```
Until i>21;
```

а) 1;

б) 21;

в) Бесконечное количество раз.

10. Чему будет равна переменная sum после выполнения фрагмента программы:

```
sum:=0;
```

```
for i:=7 to 9 do
```

```
sum:=sum+i;
```

а) 15;

б) 24;

в) 16.

11. Чему будет равна переменная sum после выполнения фрагмента программы:

```
i:=15;
```

```
sum:=0;
```

```
While i>5 do begin
```

```
sum:=sum+i/5;
```

```
i:=i-5
```

```
end;
```

а) 5;

б) 6;

в) 30.

12. Массив в программировании это...

а) переменная, которая в упорядоченном виде хранит множество значений одного типа одновременно.

б) устройство хранения данных, состоящее из нескольких накопителей информации.

в) данные с индексом.

13. Массив обозначается командным словом:

а) Arrau

б) Aray

в) Array

14. В какой строке НЕ правильное описание массива?

- а) var ch: array[1..10] of integer;
- б) var stih: array[1,5..7] of real;
- в) var STL23: array[2..108] of real;

15. Какое количество данных может находиться в данном массиве var ch: array[4..16] of integer;?

- а) 13 целых чисел;
- б) 14 целых чисел;
- в) 13 букв;

16. В данной последовательности операторов:

m:= A[1];

for i:=1 to n do

if A[i]>m then A[i]:=m else m:=A[i];

- а) ищется минимальный элемент массива A (значение m);
- б) ищется максимальный элемент массива A (значение m);
- в) меняются местами минимальный и максимальный элементы массива A;

17. Что определяет для массива X[1..n] следующий алгоритм

S:= 0;

For k:= 1 to n do

If X[k] <0 then S:=S+X[k];

- а) минимальный элемент массива;
- б) сумму отрицательных элементов массива;
- в) максимальный элемент массива;

18. Что определяет для массива X[1..n] следующий алгоритм

V:= X[1];

For k:= 1 to n do

If X[k] <V then V:=X[k];

- а) минимальный элемент массива;
- б) сумму отрицательных элементов массива;
- в) максимальный элемент массива;

20. Что определяет для массива X[1..n] следующий алгоритм

P:= 0;

For k:= 1 to n do

If X[k] <0 then P:=k;

- а) минимальный элемент массива;
- б) сумму отрицательных элементов массива;
- в) индекс последнего отрицательного элемента массива;

21. Что определяет для массива $X[1..n]$ следующий алгоритм

$R := 0;$

For $k := 1$ to n do

If $X[k] < 0$ then $R := R + 1;$

- а) минимальный элемент массива;
- б) сумму отрицательных элементов массива;
- в) количество отрицательных элементов массива;

22. Что определяет для массива $X[1..n]$ следующий алгоритм

$R := 0;$

For $k := 1$ to n do

If $X[k] < T$ then $R := R + 1;$

- а) номер первого элемента массива, большего T ;
- б) количество элементов массива, больших T ;
- в) количество элементов массива, меньших T ;

23. Что определяет для массива $X[1..n]$ следующий алгоритм

$R := n;$

For $k := 1$ to n do

If $X[k] \leq T$ then $R := R - 1;$

- а) номер первого элемента массива, большего T ;
- б) количество элементов массива, больших T ;
- в) ни один из ответов 1-2 не верен.

24. Что определяет для массива $X[1..n]$ следующий алгоритм

$P := 0;$

For $k := n$ downto 1 do

If $X[k] < T$ then $P := k;$

- а) номер первого элемента массива, большего T ;
- б) количество элементов массива, больших T ;
- в) номер первого элемента массива, меньшего T ;

25. Что определяет для массива $X[1..n]$ следующий алгоритм

$P := 0;$

For $k := 1$ to n do

If $X[k] > T$ then $P := k;$

- а) номер последнего элемента массива, меньшего T ;
- б) количество элементов массива, больших T ;
- в) последнего элемента массива, большего T ;

Образец документа об окончании курсов

По итогам обучения выдается документ следующего образца – СВИДЕТЕЛЬСТВО (макет)

Инфосфера
Автономная некоммерческая организация
дополнительного образования "Инфосфера Козьмодемьянск"

Свидетельство

№ _____

Выдано _____

в том, что он (а) с _____ 20__ г. по _____ 20__ г. обучался (ась)
в АНО ДО «Инфосфера Козьмодемьянск»
по дополнительной общеразвивающей программе
« _____ »
объемом _____ академических часов

Директор _____ Н. В. Суетенкова

г. Козьмодемьянск
« _____ » _____ 20__ г.

Лицензия на право ведения образовательной деятельности
рес №Л035-01267-12/00248265 от 22 июля 2021