

МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ «СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА №71»
Городского округа «Город Лесной» Свердловской области

Рассмотрено:

Педагогический совет

Протокол № 289

От 31.08.2021

Утверждаю

Директор МБОУ СОШ №71

И.А. Приходько

Утверждено приказом МБОУ СОШ №71

От 31.08.2021 №248-К



Дополнительная образовательная программа
«Применение информационных технологий для решения задач
по информатике»

Срок реализации: года

Возраст: 15-16 лет

Пояснительная записка

Данная программа предназначена для обучающихся общеобразовательных учреждений, желающих расширить свои знания по информатике, интересующихся данным предметом и планирующих проходить различные формы независимой оценки качества знаний. Учебная программа изучается на ступени основного общего образования.

Программа учитывает современные тенденции развития общества, наиболее значимые направления развития информационных технологий, такие как компьютерная графика, 3D-моделирование, алгоритмизация и программирование. Программа курса разработана с таким расчетом, чтобы учащиеся получили знания по информатике, необходимые для развития личности, успешного применения информационных технологий в жизни.

Необходимость создания данного курса вызвана тем, что требования к профессиям по информационным технологиям возрастают, а времени на изучение не увеличивается.

Программа курса предполагает проведение занятий в виде индивидуального и коллективного решения задач и составления алгоритмов решения задач. В процессе обучения материал излагается на теоретической и практической основах, включая вопросы теории информации, обработки и измерения информации разного вида, математических основ информатики, коммуникационных технологий, алгоритмизации и программирования, моделирования.

В процессе изучения курса, обучающиеся будут использовать систему дистанционного обучения для тренировки решения задач, контрольных работ и тестов.

Режим занятий: 1 час один раз в неделю.

Объем: курс рассчитан на 35 часов (1 час в неделю).

Срок освоения: 35 недель.

Уровневость: продвинутый уровень.

Формы обучения: фронтальная, групповая, индивидуальная.

Виды занятий: беседа, практическое занятие, тренинг, круглый стол.

Формы подведения результатов: тест, практическое занятие, контрольная работа.

Цель курса: систематизировать и расширить знания обучающихся, научить их самостоятельно составлять алгоритмы решения задач по информатике.

Задачи курса:

- приобретение определенной техники решения задач по информатике;
- анализ структуры задач;
- сформировать порядок состава операций, которые должны быть выполнены в процессе решения различных задач;
- научить основным операциям, из которых складывается процесс решения задач;

- познакомить со структурой рациональной последовательности анализа и решения задач;
- научить самостоятельному анализу задачи и подбору алгоритма её решения;
- научить переносить усвоенный алгоритм решения задач по одному разделу на другие разделы.

Учащиеся, в ходе занятий, приобретут:

- навыки самостоятельной работы;
- составлять алгоритм решения;
- проверять предлагаемые для решения гипотезы (т.е. владеть основными умственными операциями, составляющими поиск решения задачи).

Основные виды деятельности обучающихся:

Разбор проблемных ситуаций на занятиях.

Самостоятельная отработка аналогичных задач по данной теме.

Самостоятельная работа с системой дистанционного обучения.

Формы обучения: учебно-плановые (урок, лекция, домашняя работа) фронтальные, коллективные, индивидуальные, а также со сменным составом учеников.

Формы контроля знаний, умений и навыков: тесты, проекты, контрольные работы.

Учебный (тематический) план

№	Наименование раздела	Всего часов	Контр. работы
1	Теоретические основы информатики	2	
2	Алгоритмизация	8	1
3	Программирование	8	1
4	Компьютерная графика	6	1
5	3D-моделирование	11	2(1 итоговая)
	Всего за курс	35	5

Список литературы:

Абрамян М.Э. Programming Taskbook. Электронный задачник по программированию. Версия 4.6

АСКОН. Азбука КОМПАС График V15. 2014

Дергачева Л. М. Решение типовых экзаменационных задач по информатике: учебное пособие / Л. М. Дергачева. — М.: БИНОМ. Лаборатория знаний, 2013. — 360 с.: ил.

Культин Н. Б. Turbo Pascal в задачах и примерах. — СПб. БХВ-Петербург, 2006. — 256 с.: ил.

Кушниренко А. Г. Информатика. 7—9 кл.: Учеб, для общеобразоват. учеб, заведений / А. Г. Кушниренко, Г. В. Лебедев, Я. Н. Зайдельман. — 3-е изд., стереотип. — М.: Дрофа, 2002. — 336 с.: ил.

Могилев А. В. Практикум по информатике: Учеб. пособие для студ. высш. учеб. заведений / А.В.Могилев, Н.И.Пак, Е.К.Хеннер; Под ред. Е.К.Хеннера. — 2-е изд., стер. — М.: Издательский центр «Академия», 2005. — 608 с.

Уханёва В.А. Черчение и моделирование на компьютере, КОМПАС-3D LT.

Приложение

Примеры задач

1. Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **Сместиться на (a, b)** (где a, b – целые числа), перемещающую Чертёжника из точки с координатами (x, y) , в точку с координатами $(x+a, y+b)$. Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные — уменьшается.

Например, если Чертёжник находится в точке с координатами $(1, 1)$, то команда **Сместиться на (-2, 4)** переместит его в точку $(-1, 5)$.

Запись

Повтори k раз

Команда1 Команда2 Команда3

Конец

означает, что последовательность команд **Команда1 Команда2 Команда3** повторится k раз.

Чертёжнику был дан для исполнения следующий алгоритм:

Сместиться на (2, 6)

Повтори 2 раз

Сместиться на (2, 1) Сместиться на (-5, 4) Сместиться на (1,-4)

Конец

Какую команду надо выполнить Чертёжнику, чтобы вернуться в исходную точку, из которой он начал движение?

1) Сместиться на $(4, -2)$

2) Сместиться на $(-4, 2)$

3) Сместиться на $(2, -8)$

4) Сместиться на $(-2, 8)$

2. Запишите значение переменной s , полученное в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Алгоритмический язык	Бейсик	Паскаль
алг нач цел s, k $s := 0$ нц для k от 3 до 8 $s := s + 7$ кц вывод s кон	DIM k, s AS INTEGER $s = 0$ FOR $k = 3$ TO 8 $s = s + 7$ NEXT k PRINT s	Var s, k : integer; Begin $s := 0$; for $k := 3$ to 8 do $s := s + 7$; writeln(s); End.

3. В таблице Dat хранятся данные о количестве учеников в классах ($Dat[1]$ — количество учеников в первом классе, $Dat[2]$ — во втором и т. д.). Определите, какое число будет напечатано в результате работы следующей программы. Текст программы приведён на трёх языках программирования.

Алгоритмический язык	Бейсик	Паскаль
алг нач целтаб $Dat[1:11]$ цел k, m $Dat[1] := 20$ $Dat[2] := 25$ $Dat[3] := 19$	DIM $Dat(11)$ AS INTEGER DIM k, m AS INTEGER $Dat[1] = 20$ $Dat[2] = 25$ $Dat[3] = 19$ $Dat[4] = 25$ $Dat[5] = 26$	Var k, m : integer; Dat : array[1..11] of integer; Begin $Dat[1] := 20$; $Dat[2] := 25$; $Dat[3] := 19$; End.

<pre> Dat[4] := 25 Dat[5] := 26 Dat[6] := 22 Dat[7] := 24 Dat[8] := 28 Dat[9] := 26 Dat[10] := 21 Dat[11] := 27 m := 0 нц для k от 1 до 11 если Dat[k] > 22 то m := m+1 все КЦ Вывод m КОН </pre>	<pre> Dat[6] = 22 Dat[7] = 24 Dat[8] = 28 Dat[9] = 26 Dat[10] = 21 Dat[11] = 27 m = 0 FOR k := 1 TO 11 IF Dat(k) > 22 THEN m =m+1 ENDIF NEXT k PRINT m </pre>	<pre> Dat[4] := 25; Dat[5] := 26; Dat[6] := 22; Dat[7] := 24; Dat[8] := 28; Dat[9] := 26; Dat[10] := 21; Dat[11] := 27; m := 0; for k := 1 to 11 do if Dat[k] > 22 then begin m := m+1; end; writeln(m); End. </pre>
--	--	---

4. У исполнителя Квадратор две команды, которым присвоены номера:

1. возведи в квадрат

2. прибавь 1

Первая из них возводит число на экране во вторую степень, вторая — прибавляет к числу 1. Составьте алгоритм получения из числа 2 числа 27, содержащий не более 5 команд. В ответе запишите только номера команд. В ответе запишите только номера команд. (Например, 11221 — это алгоритм: возведи в квадрат, возведи в квадрат, прибавь 1, прибавь 1, возведи в квадрат, который преобразует число 2 в 324.) Если таких алгоритмов более одного, то запишите любой из них.

5. Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. У Робота есть девять команд. Четыре команды — это команды-приказы:

вверх вниз влево вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑ вниз ↓, влево ←, вправо →. Если Робот получит команду передвижения сквозь стену, то он разрушится. Также у Робота есть команда **закрасить**, при которой закрашивается клетка, в которой Робот находится в настоящий момент.

Ещё четыре команды — это команды проверки условий. Эти команды проверяют, свободен ли путь для Робота в каждом из четырёх возможных направлений:

сверху свободно снизу свободно слева свободно справа свободно

Эти команды можно использовать вместе с условием «если», имеющим следующий вид:

если условие то

последовательность команд

все

Здесь *условие* — одна из команд проверки условия. *Последовательность команд* — это одна или несколько любых команд-приказов. Например, для передвижения на одну клетку вправо, если справа нет стенки, и закрашивания клетки можно использовать такой алгоритм:

если справа свободно то

вправо

закрасить

все

В одном условии можно использовать несколько команд проверки условий, применяя логические связки **и**, **или**, **не**, например:

если (справа свободно) и (не снизу свободно) то

вправо

все

Для повторения последовательности команд можно использовать цикл **«пока»**, имеющий следующий вид:

нц пока условие

последовательность команд

кц

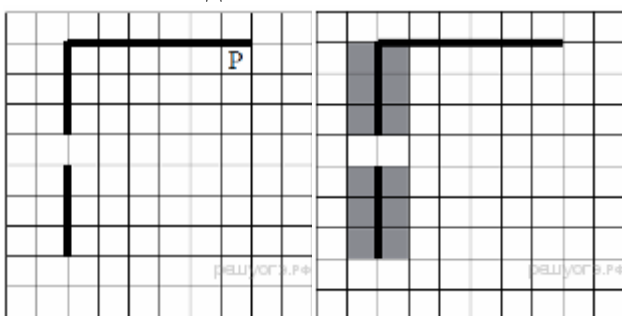
Например, для движения вправо, пока это возможно, можно использовать следующий алгоритм:

нц пока справа свободно

вправо

кц

Выполните задание.



На бесконечном поле есть горизонтальная и вертикальная стены. Левый конец горизонтальной стены соединён с верхним концом вертикальной стены. Длины стен неизвестны. В горизонтальной стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно под горизонтальной стеной у её правого конца. На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»).

Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно левее и правее вертикальной стены. Проход должен остаться незакрашенным. Робот должен закрасить только клетки,

удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок).

При исполнении алгоритма Робот не должен разрушиться, выполнение алгоритма должно завершиться. Конечное положение Робота может быть произвольным. Алгоритм должен решать задачу для любого допустимого расположения стен и любого расположения и размера проходов внутри стен. Алгоритм может быть выполнен в среде формального исполнителя или записан в текстовом редакторе. Сохраните алгоритм в текстовом файле.

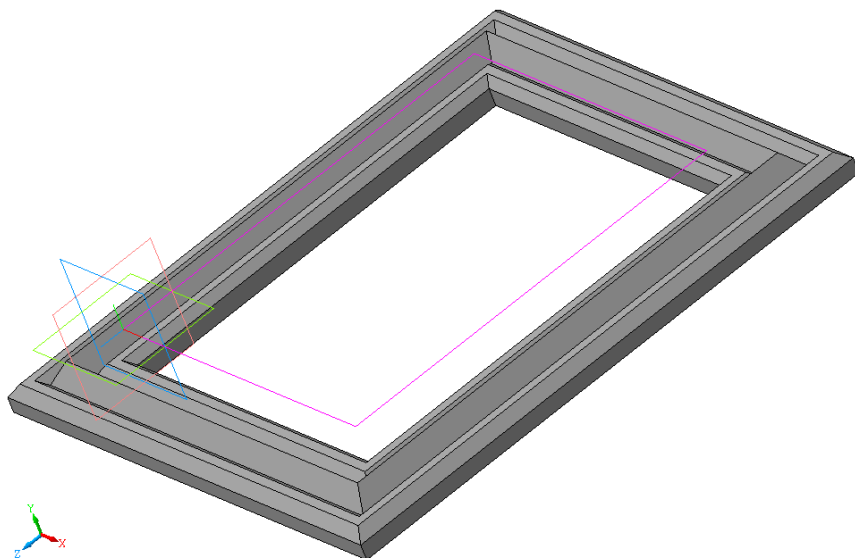
6. Напишите программу, которая в последовательности натуральных чисел определяет минимальное число, оканчивающееся на 6. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, оканчивающееся на 6. Количество чисел не превышает 1000. Введённые числа не превышают 30 000. Программа должна вывести одно число — минимальное число, оканчивающееся на 6.

Пример работы программы:

Входные данные	Выходные данные
3	16

26	
16	
36	

7. Выполните построение объемной детали в Компас-3D.



8.