

Государственное автономное профессиональное образовательное учреждение
«Тетюшский государственный колледж гражданской защиты»

УТВЕРЖДАЮ

Директор ГАПОУ

«Тетюшский государственный
колледж гражданской защиты»

Алиева Т.Ю.

Приказ № 194 от 30 июня 2025 г.



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ

наименование дисциплины

по специальности

ОП.10 Основы алгоритмизации и программирования

09.02.13. Интеграция решений с применением технологий

искусственного интеллекта

Фонд оценочных средств разработан на основе:

- федерального государственного образовательного стандарта среднего образования по специальности:

09.02.13. Интеграция решений с применением технологий искусственного интеллекта
код и наименование специальности

- рабочей программы учебной дисциплины

ОП.10 Основы алгоритмизации и программирования

наименование учебной дисциплины

- локальных актов ГАПОУ «Теплошский государственный колледж гражданской защиты»

Разработчик: Шакирова Л.И., преподаватель математики ГАПОУ «Теплошский государственный колледж гражданской защиты»

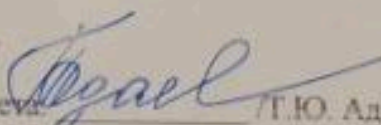
Рассмотрен и одобрен на заседании предметно-цикловой комиссии естественнонаучных дисциплин и математики ГАПОУ «Теплошский государственный колледж гражданской защиты»

протокол № 4 от 30 июня 2025 г.

председатель ПЦК:  Е.Г. Дороднова

Рассмотрен педагогическим советом ГАПОУ «Теплошский государственный колледж гражданской защиты»,

протокол № 9 от 30 июня 2025 г.

председатель педагогического совета:  Г.Ю. Адаева

1. Паспорт фонда оценочных средств по дисциплине

1.1. Общие положения

Фонд оценочных средства (ФОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины **ОП.10 Основы алгоритмизации и программирования.**

ФОС включают оценочные материалы для проведения текущего контроля успеваемости в форме практических работ, развернутых ответов (письменных), устных ответов, тестов, составления презентаций, рефератов и промежуточной аттестации в форме дифференцированного зачета.

1.2. Планируемые результаты освоения дисциплины

Содержание образовательной программы учебной дисциплины **ОП.10 Основы алгоритмизации и программирования** обеспечивает достижение студентами следующих результатов освоения дисциплины подлежащих проверке

Цели и задачи дисциплины – требования к результатам освоения дисциплины:

Целью освоения дисциплины: Разрабатывать алгоритмы для конкретных задач.

Использовать программы для графического отображения алгоритмов.

Определять сложность работы алгоритмов.

Работать в среде программирования.

Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.

Оформлять код программы в соответствии со стандартом кодирования.

Выполнять проверку, отладку кода программы.

В результате освоения дисциплины обучающийся должен:

1.3. Распределение оценивания результатов обучения

№	Наименование разделов дисциплины	Содержание раздела (темы)	Код компетенции	Знания, умения, практический опыт необходимые для формирования соответствующей компетенции	Методы оценки
1	Основы алгоритмизации	1.1. Алгоритмизация вычислительного процесса	ОК 1 ОК 2	Знать: алгоритмы выполнения работ в профессиональной и смежных областях; приемы структурирования информации; Уметь: реализовать составленный план; работать в среде программирования; реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; структурировать получаемую информацию;	Опрос по контрольным вопросам Оценка выполнения практического задания (работы)

2	Введение в программирование	2.1. Языки программирования 2.2 Типы данных	ОК 4 ОК 5 ПК 2.4	<p>Знать: основы проектной деятельности; правила оформления документов и построения устных сообщений; правила оформления документов и построения устных сообщений; модели процесса разработки программного обеспечения; приемы работы с инструментальными средствами тестирования и отладки; основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; подпрограммы, составление библиотек подпрограмм; выполнять проверку, отладку кода программы</p> <p>Уметь: организовывать работу коллектива и команды;</p> <p>грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке;</p> <p>выполнять ручное и автоматизированное тестирование программного модуля;</p> <p>Иметь практический опыт:</p> <ul style="list-style-type: none"> - разрабатывать тестовые наборы (пакеты) для программного модуля; - инспектировать разработанные программные модули на предмет соответствия стандартам кодирования. - работы в коллективе и команде 	Опрос по контрольным вопросам
---	-----------------------------	------------------------------------------------	------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------

3	<p>Объектно-ориентированное программирование</p>	<p>3.1. Основные принципы объектно-ориентированного программирования (ООП) 3.2. Интегрированная среда разработчика</p> <p>3.3. Визуальное событийно-управляемое программирование</p>	<p>ОК 9 ПК 2.5</p>	<p>Знать: современные средства и устройства информатизации;</p> <p>понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; эволюцию языков программирования, их классификацию, понятие системы программирования; основные принципы процесса разработки программного обеспечения;</p> <p>подпрограммы, составление библиотек подпрограмм; объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения Уметь: использовать современное программное обеспечение; разрабатывать алгоритмы для конкретных задач;</p> <p>использовать программы для графического отображения алгоритмов; работать в среде программирования; реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; использовать методы для получения кода с заданной функциональностью и степенью качества;</p> <p>разрабатывать алгоритмы для конкретных задач;</p> <p>Иметь практический опыт:</p> <ul style="list-style-type: none"> - разрабатывать тестовые наборы (пакеты) для программного модуля; - инспектировать разработанные программные модули на предмет соответствия стандартам кодирования. - работы в коллективе и команде 	<p>Опрос по контрольным вопросам Оценка выполнения практического задания (работы)</p>
---	--------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------

<p>4 Операторы языка программирования</p>	<p>4.1. Операторы языка программирования</p>	<p>ОК 9, ОК10, ПК 2.4. ПК 2.5</p>	<p>Знать: современные средства и устройства информатизации;</p> <p>понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; эволюцию языков программирования, их классификацию, понятие системы программирования;</p> <p>правила построения простых и сложных предложений на профессиональные темы;</p> <p>модели процесса разработки программного обеспечения; приемы работы с инструментальными средствами тестирования и отладки; основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; подпрограммы, составление библиотек подпрограмм;</p> <p>выполнять проверку, отладку кода программы;</p> <p>основные принципы процесса разработки программного обеспечения; подпрограммы, составление библиотек подпрограмм;</p> <p>объектно-ориентированную модель программирования основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения</p> <p>Уметь:</p> <p>использовать современное программное обеспечение;</p> <p>разрабатывать</p>	<p>Опрос о контроль ным вопросам Оценка выполнен ия практическ ого зада ния (работы)</p>
---------------------------------------------------	------------------------------------------------------	-----------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------

			<p>алгоритмы для конкретных задач;</p> <p>использовать программы для графического отображения алгоритмов; работать в среде программирования; реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;</p> <p>выполнять ручное и автоматизированное тестирование программного модуля;</p> <p>использовать методы для получения кода с заданной функциональностью и степенью качества;</p> <p>разрабатывать алгоритмы для конкретных задач;</p> <p>Иметь практический опыт:</p> <ul style="list-style-type: none">- разрабатывать тестовые наборы (пакеты) для программного модуля;- инспектировать разработанные программные модули на предмет соответствия стандартам кодирования.- работы в коллективе и команде	
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

<p>5 Структурное и модульное программирование</p>	<p>1. Процедуры и функции</p> <p>5.2. Структурное и модульное программирование</p>	<p>ОК 9, ОК 10,</p> <p>ПК 2.4, ПК 2.5</p>	<p>Знать: современные средства и устройства информатизации;</p> <p>понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; эволюцию языков программирования, их классификацию, понятие системы программирования;</p> <p>правила построения простых и сложных предложений на профессиональные темы;</p> <p>модели процесса разработки программного обеспечения; приемы работы с инструментальными средствами тестирования и отладки; основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; подпрограммы, составление библиотек подпрограмм;</p> <p>выполнять проверку, отладку кода программы;</p> <p>основные принципы процесса разработки программного обеспечения; подпрограммы, составление библиотек подпрограмм;</p> <p>объектно-ориентированную модель программирования; основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения</p> <p>Уметь:</p> <p>использовать современное программное обеспечение;</p> <p>разрабатывать</p>	<p>Опрос по контрольным вопросам</p> <p>Оценка выполнения практического задания (работы)</p>
---------------------------------------------------	------------------------------------------------------------------------------------	-------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------

			<p>алгоритмы для конкретных задач;</p> <p>использовать программы для графического отображения алгоритмов; работать в среде программирования; реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;</p> <p>выполнять ручное и автоматизированное тестирование программного модуля;</p> <p>использовать методы для получения кода с заданной функциональностью и степенью качества;</p> <p>разрабатывать алгоритмы для конкретных задач;</p> <p>Иметь практический опыт:</p> <p>тестировать сборки (пакеты) для программного модуля;</p> <p>инспектировать разработанные программные модули на предмет соответствия стандартам кодирования</p>	
--	--	--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

<p>6 Решение прикладных задач</p>	<p>6.1. Постановка задачи 6.2. Проектирование объектно-ориентированного приложения</p>	<p>ОК 9, ОК 10, ПК 2.4, ПК 2.5</p>	<p>Знать: современные средства и устройства информатизации; понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; эволюцию языков программирования, их классификацию, понятие системы программирования; правила построения простых и сложных предложений на профессиональные темы; модели процесса разработки программного обеспечения; приемы работы с инструментальными средствами тестирования и отладки; основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; подпрограммы, составление библиотек подпрограмм; выполнять проверку, отладку кода программы; основные принципы процесса разработки программного обеспечения; подпрограммы, составление библиотек подпрограмм; объектно-ориентированную модель программирования основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения Уметь: использовать современное программное обеспечение; разрабатывать алгоритмы</p>	<p>Опрос о контрольным вопросам Оценка выполнения практического задания (работы)</p>
-----------------------------------	--------------------------------------------------------------------------------------------	--------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------

			<p>для конкретных задач;</p> <p>использовать программы для графического отображения алгоритмов; работать в среде программирования; реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;</p> <p>выполнять ручное и автоматизированное тестирование программного модуля;</p> <p>использовать методы для получения кода с заданной функциональностью и степенью качества;</p> <p>разрабатывать алгоритмы для конкретных задач;</p> <p>Иметь практический опыт:</p> <ul style="list-style-type: none"> - разрабатывать тестовые наборы (пакеты) для программного модуля; - инспектировать разработанные программные модули на предмет соответствия стандартам кодирования. - работы в коллективе и команде 	
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

5.1 Типовые контрольные задания или иные материалы

5.1.1 Оценочные средства при текущем контроле

Текущий контроль заключается в опросе обучающихся по контрольным вопросам, выполнении практических работ.

Пример контрольных вопросов по 1 разделу:

1. Дайте определение алгоритму.
2. Свойства алгоритма.
3. Способы записи алгоритма.
4. Основные элементы блок-схемы.
5. Виды алгоритмов.
6. Отличительные особенности алгоритмов с предусловием и постусловием.

Пример контрольных вопросов по 2 разделу:

1. Опишите семантику языков программирования.
2. Чем отличаются компилируемые и интерпретируемые языки?
3. Первые языки программирования высокого уровня.
4. Алгоритмические языки программирования

Пример контрольных вопросов по 3 разделу:

1. Какие основные объекты интегрированной среды программирования вы знаете?
2. Какие стандартные объекты используются при создании прикладной программы?
3. Перечислите наиболее часто используемые свойства объектов.

Пример контрольных вопросов по 4 разделу:

1. Какие функции работы со строками вы знаете?
2. Как осуществляется сложение строк?
3. Что такое массив?
4. Какие массивы вы знаете?
5. Какие способы формирования (заполнения) массивов вы знаете?
6. Как обратиться к элементу массива?

Пример контрольных вопросов по 5 разделу:

1. Что такое подпрограмма?
2. Какие параметры называются фактическими?
3. Какие параметры называются формальными?
4. Как связаны между собой формальные и фактические параметры?
5. Какие переменные называются глобальными?
6. Какие переменные называются локальными?

Пример контрольных вопросов по 6 разделу:

1. Опишите этапы постановки задачи.
2. В чем заключается анализ, формальная постановка и выбор метода решения задачи?
3. Опишите принципы разработки алгоритмов решения задачи.
4. Опишите процесс программирования модулей, тестирования и отладки программы.

Критерии оценивания:

- 90...100 баллов – при правильном и полном ответе на все контрольные вопросы, выполнении практических работ;
- 80...89 баллов – при правильном и полном ответе на один из контрольных вопросов и правильном, но неполном ответе на другой из контрольных вопросов, выполнении практических работ;
- 60...79 баллов – при правильном и неполном ответе на два контрольных вопроса, или правильном и полном ответе только на один из контрольных вопросов, выполнении практических работ;
- 25...59 баллов – при правильном и неполном ответе только на один из контрольных вопросов, при невыполнении практических работ;
- 0...24 баллов – при отсутствии правильных ответов на контрольные вопросы, при невыполнении практических работ.

Количество баллов	0...24	25...59	60...79	80...89	90-100
Шкала оценивания	2		3	4	5

5.1.2 Оценочные средства при промежуточной аттестации

Формой промежуточной аттестации является экзамен, в процессе которого определяется сформированность обозначенных в рабочей программе компетенций. Инструментом измерения сформированности компетенций является устный ответ на 2 вопроса, наличие зачета по каждой единице текущего контроля.

Критерии оценивания:

- 90...100 баллов – при правильном и полном ответе на два вопроса;
- 80...89 баллов – при правильном и полном ответе на один из вопросов и правильном, но не полном ответе на другой из вопросов;
- 60...79 баллов – при правильном и неполном ответе на два вопроса или правильном и полном ответе только на один из вопросов;
- 0...59 баллов – при отсутствии правильных ответов на вопросы.

Количество баллов	0...59	60...79	80...89	90-100
Шкала оценивания	2	3	4	5

Вопросы к экзамену:

1. Понятие о свойствах, методах и событиях.
2. Проектирование формы. Основные свойства формы.
3. Общие для всех объектов свойства. Свойства, определяющие внешний вид объекта.
4. Общие для всех объектов свойства. Свойства, определяющие поведение объекта.
5. Общие для всех объектов свойства. Свойства группы Misc(Общие свойства).

6. Свойства позиционирования объектов.
7. Свойства шкалы размеров объекта.
8. Методы. Основные методы.
9. События. Основные события.
10. Переменные. Объявление переменных. Типы переменных. Имена переменных.
11. Область действия переменных. Присвоение значений переменным.
12. Константы. Объявление констант. Область действия констант.
13. Математические операторы (+, -, mod, \, /...).
14. Встроенные арифметические функции (ABS, FIX, CINT, INT, COS, SIN, TAN, SQR, EXP, LOG).
15. Случайные числа. Оператор RND и RANDOMIZE.
16. Управляющие конструкции. Условия IF...End IF.
17. Управляющие конструкции. Условия SELECT CASE.
18. Условные выражения. Операторы сравнения(>,<,...). Логические операции (AND, XOR, NOT, OR).
19. Циклы FOR...NEXT. Оператор EXIT.
20. Циклы DO...LOOP. Оператор EXIT.
21. Массивы. Объявление массива фиксированного размера.
22. Динамические массивы.
23. ASCII-коды. Функции ASC() и CHR().
24. Строковые данные. Функции Left, Len, LTrim...
25. Строковые данные. Функции LCase, UCase, Mid, Str, Val.
26. Программные модули.
27. Процедуры Sub. Оператор Call.
28. Процедуры Function. Оператор Call.
29. Работа с файлами. Типы файлов (последовательный,...). Операторы Open, Close.
30. Работа с файлами последовательного доступа, открытых для чтения.
31. Работа с файлами последовательного доступа, открытых для записи и для добавления.
32. Работа с файлами прямого доступа. Пользовательский тип данных. Оператор Get, Put.
33. Окно сообщений MsgBox.
34. Диалоговое окно для ввода информации InputBox.
35. Работа с датой и временем.
36. Форматирование данных.
37. Графика. Основные элементы управления для работы с графикой.
38. Графика. Графические методы Circle, Cls, Line, Point, Print, Pset. Функции RGB, QBColor.
39. Элементы управления данными. Основные свойства. Связь элементов управления данными с другими ЭУ.
40. База данных и ее структура.
41. Набор записей в БД. Навигация по набору записей. Свойства и методы объекта RecordSet.
42. Элемент управления командная кнопка. Основные свойства, методы, события ЭУ.
43. Элемент управления ярлык (надпись). Основные свойства, методы, события ЭУ.
44. Элемент управления текстовое окно. Основные свойства, методы, события ЭУ.
45. Элемент управления переключатели. Основные свойства, методы, события ЭУ.
46. Элемент управления контрольные индикаторы. Основные свойства, методы, события ЭУ.
47. Элемент управления список. Основные свойства, методы, события ЭУ.

5.2.3. Методические материалы, определяющие процедуры оценивания знаний, умений, практического опыта, характеризующие этапы формирования компетенций

Порядок организации проведения текущего контроля и промежуточной аттестации представлен в Положении о проведении текущего контроля и промежуточной аттестации обучающихся, осваивающих образовательные программы среднего профессионального образования в филиале федерального государственного образовательного учреждения высшего образования « Кузбасский государственный технический университет имени Т.Ф. Горбачева» в г. Междуреченске.

Банк тестовых заданий по дисциплине

1. Процесс, при котором функция вызывает саму себя, называется _____?
2. Какие из заголовочных файлов необходимо включить в программу на C++, чтобы можно было открыть файл для чтения в программе?
 - 1) <iostream>
 - 2) <fstream>
 - 3) <cmath>
 - 4) <time.h>
 - 5) <cstring>
3. Результатом выполнения фрагмента программы будет: _____
S:=-3;x:=1;

```
rep
eat
begi
n
s:=s
*(x
+2);
x:=
x+1
;

end;

unti
1
x<2
;
writ
e(s)
;
```

4. Укажите правильную запись вычисления функции в виде одного условного оператора:
 1. if $0 < x < 2$ then $y := \cos(2x)$ else $y := 1 - \sin(3x)$
 2. if $(x > 0)$ and $(x < 2)$ then $y := \cos(2 * x)$ else $y := 1 - \sin(3 * x)$
 3. if $(0 < x)$ and $(x < 2)$ then $y := \cos(2x)$ else $y := 1 - \sin(3x)$
 4. if $(0 < x)$ or $(x < 2)$ then $y := \cos(2 * x)$ else $y := 1 - \sin(3 * x)$
5. Раздел описания переменных начинается со служебного слова var, после которого следуют строки вида список имен переменных: тип;
Имена в списке перечисляются через запятую. Запишите один или несколько вариантов правильной записи в разделе описания переменных :
6. Оператор цикла, тело которого может не выполняться ни разу (что зависит от значения параметра цикла), – это оператор цикла:
 - 1) с предусловием (WHILE) и с постусловием (DO ... LOOP в Паскале REPEAT);
 - 2) с предусловием (WHILE) и с заданным числом повторений (FOR);
 - 3) с заданным числом повторений (FOR) и с постусловием (DO ... LOOP в Паскале REPEAT).

7. Автоматическое составление машинной программы последовательно строка за строкой исходной программы, записанной на языке программирования, выполняемое транслятором-интерпретатором НАЗЫВАЕТСЯ _____?

Языки программирования данного типа при исполнении программы за один проход переводят в машинные коды одну строку программы.

8. Чему равна сумма элементов $a[1]$ и $a[4]$ массива, сформированного следующим образом?

```
for i:=1 to 5 do a[i] := i * (i+1);
```

9. Арифметические выражения могут быть довольно сложными, поэтому при их записи должны соблюдаться правила, выберите из предложенных правил верные:

1. Все величины, входящие в арифметическое выражение, должны быть записаны в одну строку.

2. Все знаки операций должны быть обязательно проставлены;

3. Для указания порядка выполнения действий используют квадратные скобки с учетом обще принятой математической иерархии действий.

4. Нельзя писать два знака операций один за другим.

5. можно писать две константы одна за другой.

6. Нельзя писать две переменные одна за другой.

10. В операторе присваивания $summa := \text{sqr}(x) + 3 * a$ переменными являются

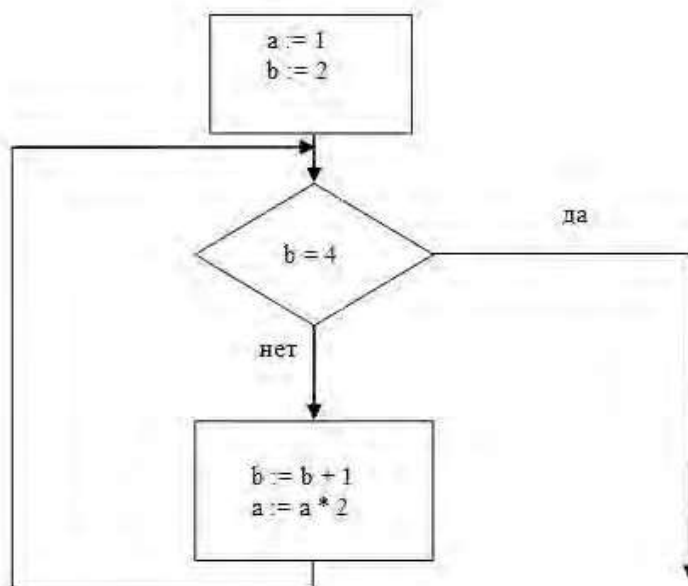
1. a, x, summa

2. x, a

3. sqr, x, a

4. summa, sqr, x, a

11. Определите значение переменной a после выполнения фрагмента алгоритма:



12. Определите сколько раз будет выполняться тело цикла:

```
S:=0;
```

```
For n:=4 downto 2 do begin
```

```
S:=s+n;
```

S:=s*10;

End;

13. Определить, что будет выведено на экран после выполнения фрагмента программы: d:=15;

Case d of

1..5: s:=15;

6..10: s:=36;

11..25: s:=30;

End;

Writeln('s = ', s);

Литература:

1. Основы алгоритмизации и программирования: учебник для среднего профессионального образования <https://urait.ru/bcode/563861> Трофимов, В. В.

2. Основы алгоритмизации и программирования на языке C# : учебник для среднего профессионального образования <https://urait.ru/bcode/565504> Кудрина, Е. В.