

Министерство образования и науки Нижегородской области
Государственное бюджетное профессиональное образовательное учреждение
«Арзамасский коммерческо-технический техникум»

УТВЕРЖДАЮ
Зам. директора по УПРиЭД
_____ А.Н. Ушанков
« ____ » _____ 20__ г.

ПРОГРАММА ТЕКУЩЕЙ АТТЕСТАЦИИ
учебной дисциплины
ОП.04 Основы алгоритмизации и программирования
по специальности среднего профессионального образования
09.02.07 Информационные системы и программирование

2022

Одобрена МО

Протокол №____

от «__» _____ 20__ г

Председатель МО:

_____ Н.И. Богомолова

Автор:

Н.А. Маликова, преподаватель информационных дисциплин ГБПОУ «Арзамасский
коммерческо-технический техникум»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели проведения текущей аттестации

При проведении текущей аттестации преподавателями должны быть достигнуты следующие цели:

- определение степени усвоения учебной дисциплины;
- стимулирование формирования практических умений и навыков, необходимых для изучения основ алгоритмизации и программирования;
- формирование готовности студентов самостоятельно применять накопленные знания при выполнении лабораторных работ;
- оценка умения использовать приобретенные знания в области программирования;
- проверка степени достижения целей учебной программы дисциплины «Основы алгоритмизации и программирования».

Формы контроля, которые необходимо выполнить обучающемуся специальности, для которых читается дисциплина

Накопление знаний (в виде информации, основ профессиональной культуры, базовых умений и навыков) обучающихся специальности 09.02.07 Информационные системы и программирование, контролируется преподавателем путем проведения следующих видов контроля:

- входного контроля;
- текущего контроля.

Ожидаемые результаты обучения

В результате изучения дисциплины «Основы алгоритмизации и программирования» обучающийся должен:

знать:

- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;
- эволюцию языков программирования, их классификацию, понятие системы программирования;
- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;
- 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:

- 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.
-

1. ТЕМАТИЧЕСКИЙ ПЛАН УЧЕБНОЙ ДИСЦИПЛИНЫ

Наименование разделов дисциплины	Кол-во часов		
	Всего часов	В том числе в форме практической подготовки	В том числе лабораторные и практические
1. Раздел 1. Введение в программирование	12	12	2
2. Раздел 2. Операторы языка программирования	40	40	18
Раздел 3. Процедуры и функции. Модульное программирование.	28	28	14
3. Раздел 4. Указатели	14	14	6
4. Раздел 5. Объектно-ориентированное программирование	84	84	40
Итого:	178	178	80

2. ВИДЫ КОНТРОЛЯ

Приобретенные обучающимися в ходе изучения дисциплины «Основы алгоритмизации и программирования» умения и знания, включающие в себя

– систему базовых знаний, отражающих общие принципы построения алгоритмов, основные алгоритмические конструкции; эволюцию языков программирования, их классификацию; понятие системы программирования; основные элементы процедурного языка программирования, структуру программы, операторы и операции, структуры данных, файлы, кассы памяти; подпрограммы, составление библиотек программ; 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;

– умения 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;

– применение на практике личного опыта использования элементов программирования в индивидуальной и коллективной учебной и познавательной, в том числе проектной деятельности контролируются преподавателем в рамках входного и текущего контроля.

2.1.Входной контроль

Входной контроль предназначен для определения степени усвоения обучающимися изучаемого теоретического материала и их готовности к отработке (выполнению) практических занятий. Входной контроль путем устного опроса осуществляет преподаватель, проводящий устный опрос. Результаты входного контроля оцениваются по двухбалльной шкале и регистрируются в рабочей тетради преподавателя.

В задания для входного контроля включаются теоретические вопросы и практические задания на основе программы по информатике и школьной программе.

Критерии оценки

Результаты входного контроля оцениваются по двухбалльной шкале и регистрируются в рабочей тетради преподавателя.

Для оценки результатов входного контроля выбраны следующие критерии:

- зачет – более 50% правильных ответов;
- незачет – менее или равно 50 % правильных ответов.

2.2. Текущий контроль

Текущий контроль предназначен для проверки качества усвоения материала по изученной теме, стимулирования своевременной учебной работы обучающихся и получения обратной связи для планирования и осуществления корректирующих и предупреждающих действий, а также, при необходимости, и коррекции методики проведения занятий.

Текущий контроль проводится в форме создания проектов, устного опроса, тестирования, самостоятельных работ, контрольных работ, лабораторных работ по вопросам, изученным как на лекциях, так и на предыдущем практическом занятии.

Результаты текущего контроля оцениваются по пятибалльной шкале и регистрируются в учебном журнале. Результаты лабораторных работ фиксируются на основной странице журнала в виде оценки и на дополнительной странице (для лабораторных работ) в виде зачёта или незачёта.

Текущий контроль может проводиться дистанционно. Задания размещаются на сайте техникума во вкладке «Студенту», раздел «Дистанционное обучение».

Контрольные материалы могут разрабатываться с помощью электронных платформ (Google Form, Onlinetestpad.com/).

Критерии оценки

Результаты текущего контроля оцениваются по пятибалльной шкале и регистрируются в учебном журнале.

Для оценки результатов текущего контроля выбраны следующие критерии.

Контрольная работа. На протяжении курса «Основы алгоритмизации и программирования» предусмотрено две контрольные работы: в третьем семестре (декабре), в четвертом семестре (апреле). Контрольная работа включает задания на составление программ. Предполагается составление программ без использования компьютера. Т.е. контрольная работа выполняется на листках.

Для оценки результатов контрольной работы выбраны следующие критерии:

- «5» (отлично) выставляется за правильное выполнение четырех предложенных заданий. Обучающийся правильно использует операторы в программе, понимает правила записи операторов; понимает, что должно получиться в результате работы программы; оценивает логику построения программы; использует знания из области математики, чтобы сравнить план решения задачи и полученные результаты. Отличная оценка предполагает самостоятельное, грамотное, правильное исполнение предложенных заданий.

- «4» (хорошо) выставляется за правильное выполнение четырех предложенных заданий. Обучающийся правильно использует операторы в программе, понимает правила записи операторов; понимает, что должно получиться в результате работы программы; оценивает логику построения программы; использует знания из области математики, чтобы сравнить план решения задачи и полученные результаты. Обучающийся выполняет задания самостоятельно, грамотно и качественно, но в выполненной работе имеются отдельные неточности. Или самостоятельно, грамотно и качественно выполнены три предложенных задания.

- «3» (удовлетворительно) выставляется, если обучающийся выполнил предложенные задания с ошибками, при этом видно, что он слабо ориентируется в программировании.

- «2» (плохо) выставляется, если обучающийся частично выполнил задания, допустив при этом ошибки, соответственно он показал разрозненные, бессистемные знания программированию. Следовательно, обучающийся, получивший – плохо, не может применять знания для составления программ.

- «1» (очень плохо) выставляется, если обучающийся не выполнил предложенные задания.

Создание проектов по заданной теме. Учитываются следующие критерии:

- 1) владение информационными технологиями: создание презентаций, сайтов, видеороликов и др.;
- 2) умение выбрать главное по заданной теме;
- 3) раскрытие актуальности темы;
- 4) полнота раскрытия темы;
- 5) грамотное исполнение проекта;
- 6) оформление, привлекательность проекта;
- 7) умение грамотно представить свой проект.

Заполняется предложенная таблица.

Фамилия, Имя обучающегося	1	2	3	4	5	6	7	Итого
Волкова Людмила	5	5	5	5	5	4	4	33
Герасимова Елена	5	4	3	3	3	3	4	25

- «5» (отлично) выставляется, если обучающийся набирает от 30 до 35 баллов
- «4» (хорошо) выставляется, если обучающийся набирает от 25 до 29 баллов
- «3» (удовлетворительно) выставляется, если обучающийся набирает от 16 до 24 баллов
- «2» (плохо) выставляется, если обучающийся набирает от 10 до 15 баллов
- «1» (очень плохо) выставляется, если обучающийся набирает ниже 10 баллов

Самостоятельная работа

Каждый обучающийся получает индивидуальное практическое задание. Устанавливается время исполнения.

Для оценки результатов самостоятельной работы выбраны следующие критерии:

- «5» (отлично) выставляется за правильное выполнение предложенного практического задания. Предполагается, что обучающийся легко ориентируется в среде программирования (Visual Studio), в которой необходимо для выполнения задания. Отличная оценка предполагает своевременное, самостоятельное, грамотное, качественное исполнение предложенных заданий.

- «4» (хорошо) выставляется за правильное выполнение предложенного практического задания. Предполагается, что обучающийся легко ориентируется в среде программирования (Visual Studio), в которой необходимо выполнять практическое задание, а также обучающийся выполняет задания своевременно, самостоятельно, грамотно и качественно, но в выполненной работе имеются отдельные неточности.

- «3» (удовлетворительно) выставляется, если обучающийся выполнил практическое задание с ошибками, при этом видно, что он слабо ориентируется в среде программирования (Visual Studio), в которой необходимо выполнять практическое задание, или решает практические задачи без нее.

- «2» (плохо) выставляется, если обучающийся частично выполнил задание, допустив при этом ошибки, соответственно он показал разрозненные, бессистемные знания по среде программирования (Visual Studio), в которой необходимо выполнять практическое задание. Следовательно, обучающийся, получивший – плохо, не может применять знания для решения практических задач.

- «1» (очень плохо) выставляется, если обучающийся не выполнил предложенное практическое задание.

Тестирование. При использовании программы MyTest возможен вариант случайных вопросов и случайных ответов. Если вопрос требует выбор одного или нескольких правильных

ответов, он оценивается в 1 балл, если вопрос предполагает введение числа или текста с клавиатуры, т.е. вопрос открыт, то он оценивается в 2 балла, если обучающемуся предложено несколько формулировок, где необходимо согласиться или не согласиться, то этот вопрос оценивается в 2 балла. Если необходимо проставить соответствие предложенных команд, терминов, то этот вопрос оценивается в 2 балла. Если необходимо проставить порядок предложенных действий, то этот вопрос оценивается в 2 балла.

Тестирование оценивается с учетом следующих критериев:

1. «5» (отлично) выставляется при условии правильного ответа обучающегося не менее чем на 85% тестовых заданий;
2. «4» (хорошо) выставляется при условии правильного ответа обучающегося не менее чем на 70% тестовых заданий;
3. «3» (удовлетворительно) – не менее чем на 55% тестовых заданий;
4. «2» (плохо) – если студент правильно ответил не менее чем на 35% тестовых заданий;
5. «1» (очень плохо) – если студент правильно ответил менее чем на 35% тестовых заданий.

3. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Перечень основной и дополнительной литературы, интернет-ресурсы

1. Семакин И.Г. Основы алгоритмизации и программирования: учебник для студ. учреждений с ред.проф.образования / И.Г. Семакин, А.П.Шестаков. - М.: Издательский центр «Академия», 2017. - 304с.

Приложение к программе текущей аттестации

Министерство образования и науки Нижегородской области
Государственное бюджетное профессиональное образовательное учреждение
«Арзамасский коммерческо-технический техникум»

УТВЕРЖДАЮ

Зам. директора по УПРиЭД

_____ А.Н. Ушанков

«__» _____ 20__ г.

**Комплект
контрольно-измерительных материалов
для текущего контроля знаний**

Специальность: 09.02.07 Информационные системы и программирование

Дисциплина: ОП.04 Основы алгоритмизации и программирования

Курс: 2

Преподаватель: (и)

_____ Н.А. Маликова

Рассмотрено на заседании МО

Протокол от «___» ___ 20__ г №___

Председатель МО

_____ Н.И. Богомолова

Входной контроль

Вопросы для устного опроса

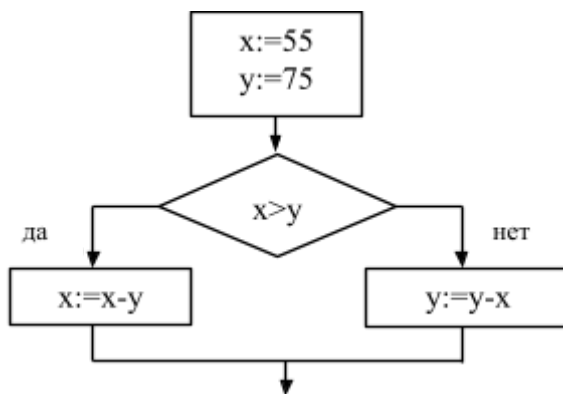
1. Алгоритмическая конструкция ветвления
2. Алгоритмическая конструкция цикла
3. Использование циклов с параметром для обработки массивов
4. Основные конструкции языка C++
5. История создания языка
6. Интегрированная инструментальная оболочка (ИИО) C++
7. Алфавит и программа на C++
8. Простые типы данных
9. Константы, переменные и оператор присваивания
10. Операторы ввода/вывода
11. Управляющие конструкции языка C++
12. Безусловные конструкции
13. Условные конструкции
14. Конструкция выбор
15. Циклические конструкции

Текущий контроль

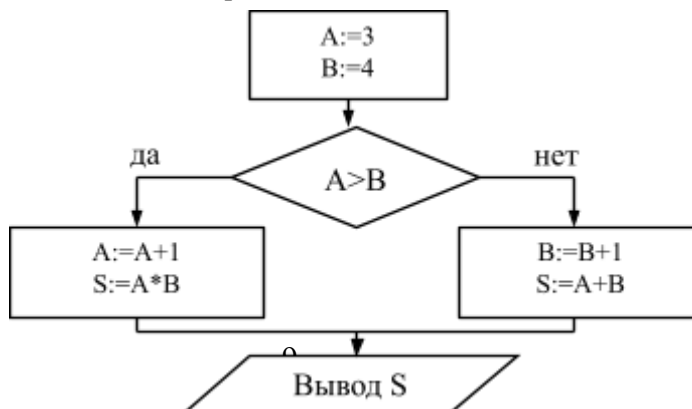
Тема 1: Основные принципы алгоритмизации и программирования

Самостоятельная работа

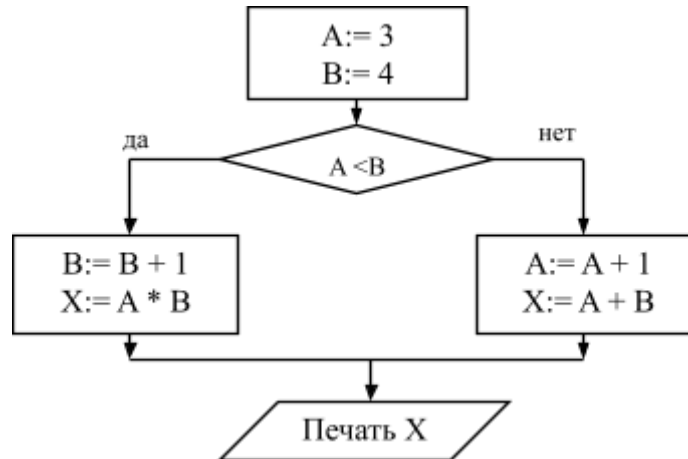
Задача 1. Фрагмент алгоритма изображен в виде блок-схемы. Определите, какое значение примет переменная x в результате выполнения этого алгоритма.



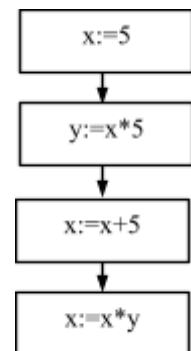
Задача 2. Фрагмент алгоритма изображен в виде блок-схемы. Определите, какое значение переменной S будет напечатано в результате выполнения алгоритма.



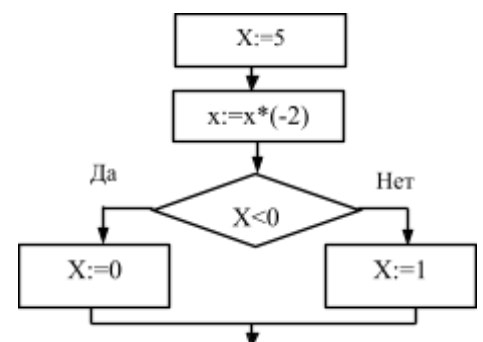
Задача 3. Фрагмент алгоритма изображен в виде блок-схемы. Определить, какое значение переменной x будет напечатано в результате выполнения алгоритма.



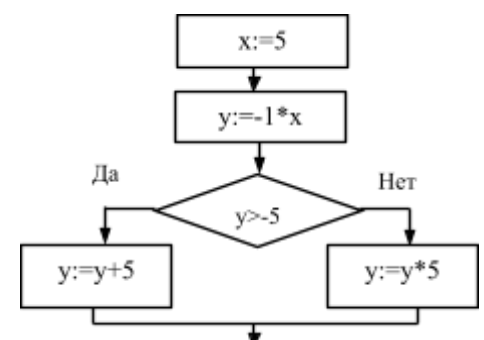
Задача 4. Фрагмент алгоритма изображен в виде блок-схемы. Определить, какое значение переменной x будет напечатано в результате выполнения алгоритма.



Задача 5. Фрагмент алгоритма изображен в виде блок-схемы. Определить, какое значение примет переменная x в результате выполнения алгоритма.



Задача 6. Фрагмент алгоритма изображен в виде блок-схемы. Определить, какое значение примет переменная y в результате выполнения алгоритма.



Тема 2: Программирование на алгоритмическом языке

Тест

1 вариант

1. Определите неправильный формат оператора While

Выберите один из 4 вариантов ответа:

- 1) While (x<10) Do cout<<"A";
- 2) While (x>10) cout<<"A";
- 3) While (x=10) cout<<"A";
- 4) While (x<=10) Do cout<<"A";

2. Определите правильный формат оператора For

Выберите один из 4 вариантов ответа:

- 1) For (i=1; i<10; i++) cout<<"A";
- 2) For (i=1; i++) cout<<"A";
- 3) For (i=1; i<10; i++) Do cout<<"A";
- 4) For (i=1; i<10) cout<<"A";

3. Определите фрагмент программы, который будет являться правильным решением неравенства

$$U = \begin{cases} \ln x, & \text{если } x > 1 \\ 3x^2 - 5x, & \text{если } x \leq 1 \end{cases}$$

Выберите несколько из 5 вариантов ответа:

- 1) If (x>1) U=ln(x);
- 4) If (x>1) U=3*x*x-5*x;
- If (x<=1) U=3*x*x-5*x;
- If (x<=1) U=ln(x);

2) If (x>1) U=ln(x); else

5) If (x>1) U=ln x else U=3*x*x-5*x;

U=3*x*x-5*x;

3) If (x<1) U=ln(x);

If (x>=1) U=3*x*x-5*x;

4. Очистку экрана осуществляет команда ...

Выберите один из 5 вариантов ответа:

- 1) ClrScr
- 2) Crt
- 3) Begin
- 4) Window
- 5) Repeat

5. Определите соответствие

Укажите соответствие для всех 6 вариантов ответа:

- | | |
|---|-------------|
| 1) Очистка экрана | ___ For ... |
| 2) Циклический оператор (цикл с предусловием) | ___ Uses |
| 3) Переменные | ___ Const |
| 4) Постоянные | ___ ClrScr |
| 5) Подключение дополнительных модулей | ___ Var |
| 6) Циклический оператор (цикл с параметром) | ___ While |

2 вариант

1. Установите порядок следования команд для вычисления выражения $R = 5 \sin x - \operatorname{tg}(1/x)$

Укажите порядок следования всех 6 вариантов ответа:

1. $R = 5 * \sin(x) - \sin(1/x) / \cos(1/x)$
2. $\operatorname{cin} >> x$
3. $\operatorname{cout} << "R=" << R$
4. $\{ 5. \}$
6. `double x, r;`

2. Определите ошибочную запись фрагмента программы

Выберите несколько из 5 вариантов ответа:

- 1) If (x>1) U=ln(x)
- 2) If (x>1) U=ln(x); else U=3*x*x-5*x
- 3) If (x>=1) U=3*x*x-5*x
- 4) If (1<x<=1) U=3*x*x-5*x
- 5) If (x>1) and (x<=-1) U=ln x else U=3*x*x-5*x

3. Определите значение U, после выполнения фрагмента программ

If ($x > 1$) $U = x * x - 3$;
 If ($x \leq 1$) $U = 3 * x * x - 5 * x$;
 при условии, что $x = 5$
Запишите число:

4. Определите значение U , после выполнения фрагмента программ

If ($x > 1$) $U = x * x - 3$
 If ($x \leq 1$) $U = 3 * x * x - 5 * x$
 при условии, что $x = 0$

Запишите число:

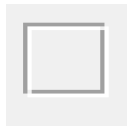
5. Определите соответствие

Укажите соответствие для всех 7 вариантов ответа:

1) Циклический оператор (цикл с постусловием)	___ Uses
2) Вывод	___ ReadLn
3) Ввод данных	___ While
4) Подключение дополнительных модулей	... Do ___ WriteLn
5) Циклический оператор (цикл с предусловием)	___ Repeat ... Until

Тема 2: Программирование в объектно-ориентированной среде

Тест



1. Запишите название объекта

Запишите ответ: _____

2. Для изменения месторасположения формы на экране используем следующий код:

Выберите один из 5 вариантов ответа:

- 1) `Form1.Caption :=`
- 2) `Form1.Width :=`
- 3) `Form1.Color :=`
- 4) `Form1.Left :=`
- 5) `Form1.Height :=`

3. Для изменения размера формы используем следующий код:

Выберите один из 5 вариантов ответа:

- 1) `Form1.Caption :=`
- 2) `Form1.AutoSize :=`
- 3) `Form1.Left :=`
- 4) `Form1.Width :=`
- 5) `Form1.Top :=`

4. Объекты, расположенные на вкладке Standard

Выберите несколько из 6 вариантов ответа:

- 1) ListBox
- 2) CheckBox
- 3) RadioButton
- 4) Timer
- 5) ScrollBar
- 6) CheckListBox

5. Объект Button находится на вкладке:

Выберите один из 4 вариантов ответа:

- 1) System
- 2) Standard
- 3) Additional

4) Win32.

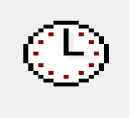
6. Для изменения шрифта текста в поле ввода используем следующий код:

Выберите один из 5 вариантов ответа:

- 1) Edit.Top :=
- 2) Form1.Top :=
- 3) Edit.Font :=
- 4) Label.Caption :=
- 5) Label.Font :=

7. Поставьте соответствие между названиями компонентами и их обозначениями

Укажите соответствие для всех 5 вариантов ответа:

- 1)  2)  3)  4)  5) 

- ___ RadioGroup
- ___ Timer
- ___ Image
- ___ RadioButton
- ___ Shape

8. Поле редактирования - ...

Выберите один из 4 вариантов ответа:

- 1) Button
- 2) Caption
- 3) Label
- 4) Edit

9. Поставьте соответствие между компонентами и их свойствами

Укажите соответствие для всех 6 вариантов ответа:

- 1) Position
- 2) Items
- 3) Text
- 4) Interval
- 5) Checked
- 6) Picture
- ___ ListBox
- ___ TrackBar
- ___ Edit
- ___ Image
- ___ RadioButton
- ___ Timer

10. Для изменения названия формы используем следующий код:

Выберите один из 4 вариантов ответа:

- 1) Form1.Name := 'Мояформа'
- 2) Form1.Text := 'Мояформа'
- 3) Form1.Form := 'Мояформа'
- 4) Form1.Caption := 'Мояформа'

Ключ к тесту

- 1) (2 б.) Верный ответ: "Panel".
- 2) (1 б.) Верные ответы: 4;
- 3) (1 б.) Верные ответы: 4;
- 4) (2 б.) Верные ответы: 1; 2; 3; 5;
- 5) (1 б.) Верные ответы: 2;
- 6) (1 б.) Верные ответы: 3;
- 7) (3 б.) Верные ответы:
 - 3;
 - 4;
 - 5;

2;
1;
8) (1 б.) Верные ответы: 4;
9) (5 б.) Верные ответы:

2;
1;
3;
6;
5;
4;

10) (1 б.) Верные ответы: 4;

Самостоятельная работа

Задание.

Проанализируйте код и опишите, какие происходят действия в результате ее работы, с какими объектами и в результате, каких событий.

```
#include <iostream>
#include <cmath>
using namespace std;
double func1(double x, double y);
double func2(double x, double y);
double func3(double x, double y);
int main()
{
    setlocale(LC_ALL, "Rus");
    double x, y;
    double f1, f2, f3;
    cout << "Введите число x: ";
    cin >> x;
    cout << "Введите число y: ";
    cin >> y;
    f1 = func1(x, y);
    f2 = func2(x, y);
    f3 = func3(x, y);
    cout << "\nПервая функция = "<<f1;
    cout << "\nВторая функция = "<<f2;
    cout << "\nТретья функция = "<<f3;
    cout << endl;
    return 0;
}
double func1(double x, double y)
{
    double n, v, c, z;
    n = x * log(x);
    v = y;
    c = cos(x) - x / 3;
    z = n + v / c;
    return z;
}
double func2(double x, double y)
{
    double d, f, g, h;
```

```

    d = x;
    f = pow(10, sin(x));
    g = cos(x - y);
    h = d - f + g;
    return h;
}
double func3(double x, double y)
{
    double a, b, s;
    a = pow(1 + 1 / pow(x, 2), x);
    b = 12 * pow(x, 2) * y;
    s = a - b;
    return s;
}

```

Контрольная работа 1 вариант

1. Интегрированная среда разработки Visual Studio. Назначение, возможности, интерфейс программы.
2. Графические операторы в Visual Studio.
3. Назначение свойств Top, Caption, Name, Height, ClientWidth, Align, Visible, Приведите примеры использования в программе.
4. На форму поместить компонент StringGrid. Создать обработчик onCreate формы, в результате которого формируется таблица, размером 10 строк и 15 столбцов. Ширина и высота ячеек = 40. Таблица заполняется произвольными числами.

2 вариант

1. Этапы разработки приложения. Охарактеризуйте один из этапов.
2. Объект StringGrid в Visual Studio. Основные свойства.
3. Назначение свойств Left, Text, Width, Height, ClientWidth, WordWrap, AutoSize. Приведите примеры использования этих свойств в программе.
4. Рассчитать средний балл успеваемости вашей группы по дисциплине, если балл – выше 4, то форма окрашивается в красный цвет, если - от 3 до 4 – в синий, иначе - в зеленый.

Проекты

1. Расписание занятий
2. Построение математических графиков
3. Работа с одномерными и двумерными массивами