Министерство образования Иркутской области
Областное государственное образовательное
учреждение среднего профессионального образования
«Иркутский авиационный техникум»

УТВЕРЖДАЮ
Директор
ОГБОУ СПО "ИАТ"

_____/Семёнов В.Г.
«29» мая 2015 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

ОП.09 Основы алгоритмизации и программирования

специальности

09.02.01 Компьютерные системы и комплексы

Иркутск, 2015

Рассмотрена
цикловой комиссией

Председатель ЦК


/А.А. Белова /

№	Разработчик ФИО
1	Филимонова Ольга Николаевна

1. ОБЩИЕ ПОЛОЖЕНИЯ

1.1. Область применения фонда оценочных средств (ФОС)

ФОС по дисциплине является частью программы подготовки специалистов среднего звена по специальности 09.02.01 Компьютерные системы и комплексы

1.2. Место дисциплины в структуре ППССЗ:

ОП.00 Общепрофессиональный цикл.

1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины

В результате освоения дисциплины обучающийся должен	№ дидактической единицы	Формируемая дидактическая единица
Знать	1.1	общие принципы построения и использования языков программирования, их классификацию;
	1.2	современные интегрированные среды разработки программ;
	1.3	процесс создания программ;
	1.4	стандарты языков программирования;
	1.5	общую характеристику языков ассемблера; назначение, принципы построения и использования
	1.6	базовые структуры данных, их преимущества и недостатки, алгоритмы работы с базовыми структурами данных;
Уметь	2.1	формализовать поставленную задачу;
	2.2	применять полученные знания к различным предметным областям;
	2.3	составлять и оформлять программы на языках программирования;
	2.4	тестировать и отлаживать программы;
	2.5	формализировать прикладные задачи и интерпретировать их в терминах программирования;

1.4. Формируемые компетенции:

ОК.1 Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.
ОК.2 Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.
ОК.3 Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.
ОК.4 Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.
ОК.5 Использовать информационно-коммуникационные технологии в профессиональной деятельности.
ОК.6 Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.
ОК.7 Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.
ОК.8 Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.
ОК.9 Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

ПК.2.1 Создавать программы на языке ассемблера для микропроцессорных систем.
ПК.2.3 Осуществлять установку и конфигурирование персональных компьютеров и подключение периферийных устройств.
ПК.3.3 Принимать участие в отладке и технических испытаниях компьютерных систем и комплексов, инсталляции, конфигурировании программного обеспечения.

2. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

2.1 Текущий контроль (ТК) № 1

Тема занятия: 2.1.1.Элементы языка С++. Идентификаторы. Служебные слова. Структура программы

Метод и форма контроля: Контрольная работа (Опрос)

Вид контроля: Опрос

Дидактическая единица: 1.1 общие принципы построения и использования языков программирования, их классификацию;

Занятие(-я):

1.1.3.Элементы языков программирования. Системы программирования. Среды программирования. Структуры программ

Задание №1

Заполните таблицу "Классификация языков программирования" недостающими данными:

Факторы	**Характеристика**	**Группы**	**Примеры ЯП**
Уровень ЯП	Степень близости ЯП к архитектуре компьютера	Низкий	
		Высокий	
		Сверхвысокий	Сетл
Специализация ЯП	Потенциальная или реальная область применения		Algol, Pascal, Basic Fortran, Cobol, Ada
		Процедурные	
		Непроцедурные	

Оценка	***Показатели оценки***

<table>
<tr><td rowspan="6">5</td><td>Факторы</td><td>Характеристика</td><td>Группы</td><td>Примеры ЯП</td></tr>
<tr><td>Уровень ЯП</td><td>Степень близости ЯП к архитектуре компьютера</td><td>Низкий
Высокий</td><td>Ассемблер
Pascal,
Basic,
Fortran, C…</td></tr>
<tr><td></td><td></td><td>Сверхвысокий</td><td>Сетл</td></tr>
<tr><td>Специализация ЯП</td><td>Потенциальная или реальная область применения</td><td>Общего назначения
Специализированные</td><td>Algol, Pascal, Basic
Fortran, Cobol, Ada</td></tr>
<tr><td>Алгоритмичность</td><td>Возможность абстрагироваться от деталей алгоритма</td><td>Процедурные

Непроцедурные</td><td>Ассемблер,
Pascal,
Basic, Fortran
Prolog</td></tr>
<tr><td colspan="4"></td></tr>
<tr><td>4</td><td colspan="4">Таблица заполнена не полностью (6 ячеек), допущены ошибки</td></tr>
<tr><td>3</td><td colspan="4">Таблица заполнена не полностью (4 ячейки), допущены ошибки</td></tr>
</table>

Задание №2

Перечислите элементы языков программирования, и дайте определение каждому элементу.

Оценка	***Показатели оценки***
5	Перечислены следующие элементы, даны все определения: Алфавит - совокупность символов, отображаемых на устройствах печати и экрана и/или вводимых с клавиатуры терминала. Лексика - совокупность правил образования цепочек символов, образующих идентификаторы (переменные и метки), операторы, операции и другие лексические компаненты языка. Синтаксис - совокупность правил образования языковых конструкций, или предложения ЯП - блоков, процедур, составных операторов, опрераторов цикла, условных операторов и пр. Особенностью синтаксиса является принцип вложенности (рекурсивность)правил построения конструкций. Семантика - смысловое содержание конструкций, предложений языка, семантический анализ - это проверка смысловой правильности конструкции.

4	Перечислены все элементы, даны три определения. В определениях допущены неточности.
3	Перечислены все элементы, даны два определения. В определениях допущены неточности

Задание №3

Перечислите из каких частей состоит исходная прграмма, дайте определение каждой из частей.

Оценка	***Показатели оценки***
5	Перечислены все основные части, даны определения. 1. Раздел идентификации - раздел, содержащий наименование программы; 2. Раздел данных - идентификация переменных (объявление, писание); 3. Раздел процедур - программная часть, содержащая описание процессов обработки данных.
4	Перечислены все основные части, определения даны не все, или допущены неточности.
3	Перечислены не все основные части, в определениях допущены неточности.

Дидактическая единица: 1.6 базовые структуры данных, их преимущества и недостатки, алгоритмы работы с базовыми структурами данных;

Занятие(-я):

1.1.1.Понятие алгоритма. Свойства алгоритмов. Формы записи алгоритмов. Запись алгоритмов в виде блок-схем

1.1.2.Типы данных

Задание №1

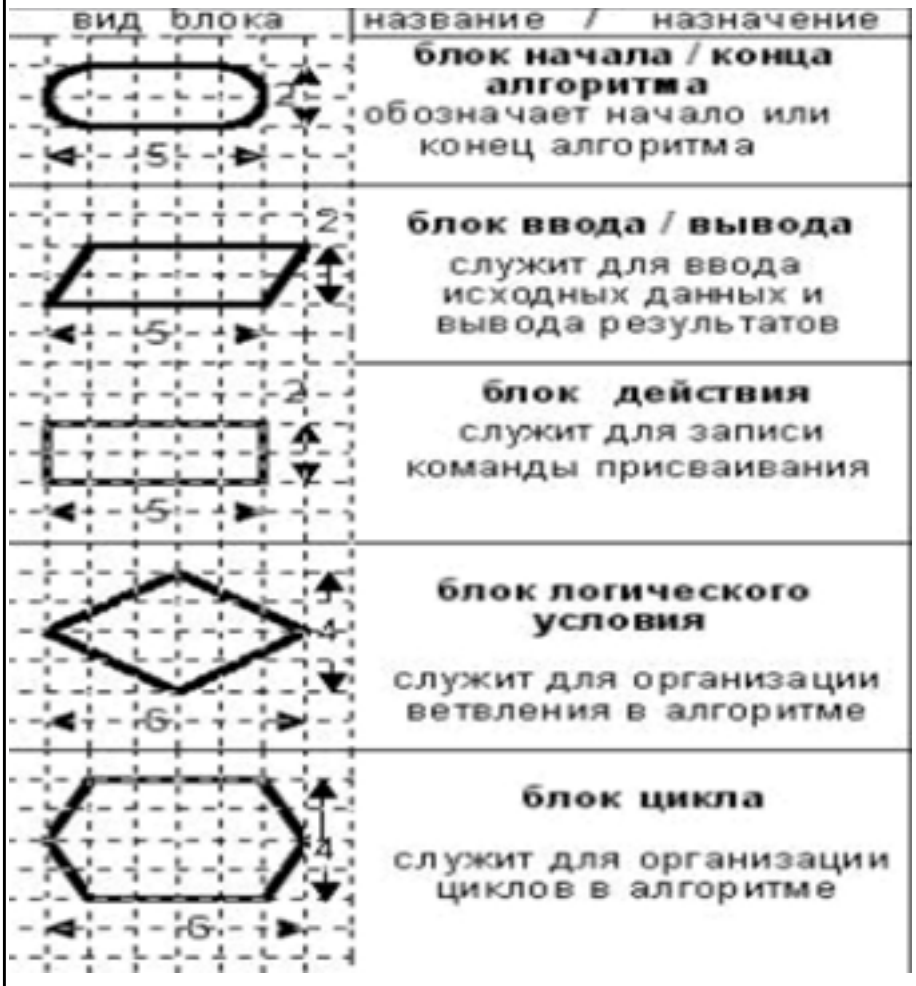
Дайте определение понятию "Алгоритм";

Перечислите и объясните свойства алгоритма;

Назовите формы записи алгоритмов;

Назовите и изобразите 5 основных элементов блок-схем.

Оценка	***Показатели оценки***

5	Алгоритм - система четких однозначных указаний, которая определяет последовательность действий над некоторыми объектами и после конечного числа шагов приводит к плучению требуемого результата. Свойства алгоритмов: 1. Дискретность - разделение выполнения решения задачи на отдельные операции. 2. Определенность - каждая команда определяет однозначное действие. 3. Понятность - система команд понятна исполнителю. 4. Результативность - исполнение алгоритма должно закончиться за конечное число шагов. Формы записи алгоритма: - словесная; - графическая; - псевдокоды (алгоритмический язык); - программная. Основные элементы блок-схем: 
4	На один из вопросов не дан ответ / допущены ошибки в формулировке ответа

3	На два вопроса не даны ответы / на один вопрос не дан ответ и допущены ошибки в формулировке ответа

Дидактическая единица: 2.1 формализовать поставленную задачу;
Занятие(-я):
1.1.1.Понятие алгоритма. Свойства алгоритмов. Формы записи алгоритмов. Запись алгоритмов в виде блок-схем
Задание №1
Построить алгоритм решения задач в виде блок-схем:
1. Дано значение температуры T в градусах Фаренгейта. Определить значение этой же температуры в градусах Цельсия. Температура по Цельсию TC и температура по Фаренгейту TF связаны следующим соотношением: TC = (TF − 32)·5/9.
2. Известно, что X кг конфет стоит A рублей. Определить, сколько стоит 1 кг и Y кг этих же конфет.
3. Поменять местами содержимое переменных A и B и вывести новые значения A и B.
4. Для данного вещественного x найти значение следующей функции f, принимающей вещественные значения:

$$f(x) = \begin{cases} 2\cdot\sin(x), & \text{если } x > 0, \\ 6 - x, & \text{если } x \le 0. \end{cases}$$

5. Даны целые числа K и N ($N > 0$). Вывести N раз число K.

Оценка	***Показатели оценки***
5	Алгоритм построен верно для всех задач
4	Алгоритм не построен для одной задачи / Допущены ошибки в построении блок-схемы.
3	Алгоритм не построен для двух задач / Алгоритм не построен для одной задачи и допущены ошибки в построении блок-схем.

2.2 Текущий контроль (ТК) № 2

Тема занятия: 2.1.12.Программирование и отладка индивидуального задания
Метод и форма контроля: Практическая работа (Опрос)
Вид контроля: Защита
Дидактическая единица: 1.2 современные интегрированные среды разработки программ;
Занятие(-я):
1.1.3.Элементы языков программирования. Системы программирования. Среды программирования. Структуры программ

2.1.1.Элементы языка C++. Идентификаторы. Служебные слова. Структура программы
2.1.2.Типы данных. Объявления переменных и констант. Преобразования типов данных
2.1.4.Операции и выражения. Приоритеты операций
2.1.6.Операторы языка
2.1.7.Условный оператор. Составление алгоритмов с логическими условиями
2.1.9.Оператор выбора (переключатель). Составление алгоритмов с использованием оператора выбора
2.1.11.Операторы цикла. Составление алгоритмов с использованием операторов цикла

Задание №1

Ответьте на следующие вопросы:

Вариант №1

1. После имени функции ставятся _____.
2. В чем особенность функции main ()?
3. Напишите оператор, выводящий на экран
    1. число 509;
    2. значение переменной x.
4. Напишите оператор, который получает с клавиатуры числовое значение и присваивает его переменной temp?
5. На какую величину увеличивает значение переменной операция инкримента?
6. Какой знак в языке C++ является разделителем и отмечает конец инструкции?
7. Каково назначение директивы # include в языке C++?
8. Как производится объявление константы в языке C++? Объявите константу Pi=3,14.
9. Каково назначение функции getch () в C++?
10. Каким образом осуществляется ввод/вывод данных?

Вариант №2

1. Тело функции заключается в _____.
2. Истинно ли следующее утверждение: в операции присваивания величина, стоящая слева от знака равенства, всегда равна величине, стоящей справа от знака равенства?
3. Запишите на языке C++ ab.

4. Какой заголовочный файл нужно включить в исходный текст, чтобы использовать объекты cin и count?
5. Напишите оператор, увеличивающий значение переменной temp на 23 с одновременным присваиванием. Напишите аналогичный оператор, не использующий сложения с присваиванием.
6. Какие значения выведут на экран два указанных оператора, если начальное значение переменной w равно 20?

    1. Cout << w --;
    2. Cout << ++ w;
7. С какой целью заголовочные файлы включаются в тексты программ?
8. Перечислите основные типы данных в C++?
9. Каково назначение объектов cin / cout?
10. Как производится объявление переменной в языке C++?

Оценка	***Показатели оценки***
3	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 5-6
4	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 7-8
5	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 9-10

Дидактическая единица: 1.3 процесс создания программ;
Занятие(-я):
1.1.3.Элементы языков программирования. Системы программирования. Среды программирования. Структуры программ
2.1.1.Элементы языка C++. Идентификаторы. Служебные слова. Структура программы
2.1.2.Типы данных. Объявления переменных и констант. Преобразования типов данных
2.1.4.Операции и выражения. Приоритеты операций
2.1.6.Операторы языка
2.1.7.Условный оператор. Составление алгоритмов с логическими условиями
2.1.9.Оператор выбора (переключатель). Составление алгоритмов с использованием оператора выбора
2.1.11.Операторы цикла. Составление алгоритмов с использованием операторов

цикла

Задание №1

1. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << a << "=Z(" << b << ")";
```

Здесь a и b — целые переменные.

2. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << "a,=Z(" << b << ")";
```

Здесь a и b — целые переменные.

3. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << "Z(a)=(b)";
```

Здесь a и b — целые переменные.

4. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << "Z(" << a << ")=(" << b << ")";
```

Здесь a и b — целые переменные.

5. Запишите оператор для вывода значений целых переменных a=5 и b=3 в потокcout в следующем формате:

```
5+3=?
```

6. Запишите оператор для вывода значений целых переменных a=5 и b=3 в поток cout в следующем формате:

```
Z(5)=F(3)
```

7. Запишите оператор для вывода значений целых переменных a=5 и b=3 в поток cout в следующем формате:

```
a=5;b=3;
```

8. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 5 ) a = 5;
```

9. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
```

```
if ( a > 5 ) a = 12;
```

10. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 5 )
a = a + 12;
else
a = a - 7;
```

11. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 15 )
a = a + 12;
else
a = a - 7;
```

12. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 5 ) a = a + 12;
else a = a - 7;
if ( a < 5 ) a = a + 12;
else a = a - 7;
```

13. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a > 5 ) a = a + 12;
else a = a - 7;
if ( a > 5 ) a = a + 12;
else a = a - 7;
```

14. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10; b = 3;
if ( a > b ) b = b + 12;
else a = a - 7;
if ( a > b ) a = a + 12;
else a = a - 7;
```

Оценка	***Показатели оценки***
3	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 6-8

4	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 9-12
5	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 13-14

Дидактическая единица: 1.4 стандарты языков программирования;
Занятие(-я):
2.1.1.Элементы языка С++. Идентификаторы. Служебные слова. Структура программы
2.1.2.Типы данных. Объявления переменных и констант. Преобразования типов данных
2.1.4.Операции и выражения. Приоритеты операций
2.1.6.Операторы языка
2.1.7.Условный оператор. Составление алгоритмов с логическими условиями
2.1.9.Оператор выбора (переключатель). Составление алгоритмов с использованием оператора выбора
2.1.11.Операторы цикла. Составление алгоритмов с использованием операторов цикла
Задание №1

1. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 2 раза?

```
 i = 4;
 while ( i <= ... ) {
 cout << "Привет!\n";
 i++;
 }
```

2. Сколько раз будет выполнен этот цикл?

```
 i = 6;
 while ( i < 5 ) {
 cout << "Привет!\n";
 i++;
 }
```

3. Какое число будет выведено на экран в результате выполнения этого цикла?

```
 i = 5;
 cout << "9";
```

```
while ( i < 5 ) {
cout << i;
i++;
}
```

4. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 4 раза?

```
i = 3;
while ( i >= ... ) {
cout << "Привет!\n";
i--;
}
```

5. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 2 раза?

```
i = ...;
while ( i >= 2 ) {
cout << "Привет!\n";
i--;
}
```

6. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 2 раза?

```
i = ...;
while ( i <= 2 ) {
cout << "Привет!\n";
i++;
}
```

7. Чему будет равно значение переменной «a» после выполнения этого цикла?

```
i = 4;
a = 12;
while ( i < 5 ) {
a += i;
i++;
}
```

8. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 10;
```

```
for( i=1; i<=2; i++ )
a -= 1;
```

9. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 10;
for ( i=0; i<=3; i++ )
a -= i;
```

10. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 2;
for ( i=0; i<=3; i++ )
a += i;
```

11. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 4;
for ( i=1; i<=4; i++ )
a += a;
```

12. Какое число будет выведено на экран после выполнения этого фрагмента программы?

```
for ( i=2; i>=0; i-- )
cout << i;
```

13. Какое число будет выведено на экран после выполнения этого фрагмента программы?

```
cout << 4;
for ( i=4; i>=3; i-- )
cout << 2*i;
```

14. Какое число будет выведено на экран после выполнения этого фрагмента программы?

```
cout << 1;
for ( i=2; i<=0; i++ )
cout << i;
```

Оценка	***Показатели оценки***

3	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 6-8
4	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 9-12
5	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 13-14

Дидактическая единица: 2.3 составлять и оформлять программы на языках программирования;

Занятие(-я):

2.1.3.Программирование простейших алгоритмов

2.1.5.Операции, выражения, приоритеты операций

2.1.8.Программирование и отладка индивидуального задания

2.1.10.Программирование и отладка индивидуального задания

Задание №1

Написать программу, которая выводит на экран фразу "Каждый охотник желает знать, где сидят фазаны", позволяющую запомнить порядок следования цветов радуги (первая буква слова кодирует цвет: каждый — красный, охотник — оранжевый, желает — желтый, знать — зеленый, где — голубой, сидят — синий, фазаны — фиолетовый). Каждое слово фразы должно быть выведено наиболее подходящим цветом.

Оценка	***Показатели оценки***
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

Написать программу вычисления сопротивления электрической цепи, состоящей из двух параллельно соединенных сопротивлений. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
?????????? ???????????? ???????????? ??????? ???????????
?????????? ?????????.??????? ???????? ??????:???????? ??????
? ???????????? (??) -> 15
```

???????? ??????? ????
????????? (??) -> **20**???????????? ????: *8.57 ??*

Оценка	***Показатели оценки***
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры
5	Успешный запуск компилятора

Дидактическая единица: 2.5 формализировать прикладные задачи и интерпретировать их в терминах программирования;
Занятие(-я):
2.1.3.Программирование простейших алгоритмов
2.1.5.Операции, выражения, приоритеты операций
2.1.8.Программирование и отладка индивидуального задания
2.1.10.Программирование и отладка индивидуального задания
Задание №1
Написать программу, которая вычисляет величину тока, потребляемого электроприбором (I = P/U, где: I — ток, А; P — мощность, Вт; U — напряжение, В). Программа должна проверять правильность введенных пользователем данных и, если они неверные (делитель равен нулю), выводить сообщение об ошибке. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

??? ? ????????????
????????????, ?? -> **60**??????????, ? -> **0**
??????! ?????????? ?? ?????? ???? ????? ????.

Оценка	***Показатели оценки***
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2
Написать программу, которая преобразует введенное пользователем десятичное число в двоичное. Рекомендуемый вид экрана программы приведен ниже.

```
????????????? ?????????? ????? ? ????????????? ????? ???
?? ?? 0 ?? 255 ? ??????? <
Enter>->49
?????????? ????? 49 ???????????? ???????? 0011000??? ????
?????? ??????? <Enter>
```

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

2.3 Текущий контроль (ТК) № 3

Тема занятия: 2.1.20.Программирование и отладка индивидуального задания
Метод и форма контроля: Практическая работа (Опрос)
Вид контроля: Защита
Дидактическая единица: 2.3 составлять и оформлять программы на языках программирования;
Занятие(-я):
2.1.12.Программирование и отладка индивидуального задания
2.1.14.Программирование и отладка индивидуального задания
2.1.16.Программирование и отладка индивидуального задания
2.1.18.Программирование и отладка индивидуального задания

Задание №1

Написать программу, которая из введенного в одной строке полного имени человека выделяет имя, отчество и фамилию. Рекомендуемый вид экрана программы приведен ниже (данные, введенные пользователем, выделены полужирным).

```
??????? ? ?
???? ?????? ???, ?????
??? ? ???????-> ???? ???????? ??????
???: ???????????: ??????????????: ????????? ?????????? ???
???? <Enter>
```

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Дидактическая единица: 2.4 тестировать и отлаживать программы;
Занятие(-я):
2.1.3.Программирование простейших алгоритмов
2.1.5.Операции, выражения, приоритеты операций
2.1.8.Программирование и отладка индивидуального задания
2.1.10.Программирование и отладка индивидуального задания
2.1.12.Программирование и отладка индивидуального задания
2.1.14.Программирование и отладка индивидуального задания
2.1.16.Программирование и отладка индивидуального задания
2.1.18.Программирование и отладка индивидуального задания

Задание №1

Написать *функцию*, которая вычисляет сопротивление цепи, состоящей из двух резисторов, которые могут быть соединены последовательно или параллельно. Функция должна проверять корректность параметров: если неверно указан тип соединения, то функция должна возвращать –1.

Оценка	***Показатели оценки***
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

Написать программу, которая дописывает в находящийся на диске компьютера файл contacts.txt имя, фамилию и номер телефона, например, вашего товарища. Если файла на диске нет, то программа должна создать его. В файле каждый элемент данных (имя, фамилия, телефон) должен находиться в отдельной строке. Ниже приведен рекомендуемый вид экрана во время работы программы.

```
????
?????? ?
????????? ? ????
?????? ???????????????? -> ?????????? -> ????
??????? -> 234-84-37
?????????? ??????????? ?????????? ??????? <Enter>
```

Оценка	***Показатели оценки***
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы

5	Успешный запуск компилятора

Дидактическая единица: 2.2 применять полученные знания к различным предметным областям;

Занятие(-я):

2.1.3.Программирование простейших алгоритмов
2.1.5.Операции, выражения, приоритеты операций
2.1.8.Программирование и отладка индивидуального задания
2.1.10.Программирование и отладка индивидуального задания
2.1.12.Программирование и отладка индивидуального задания
2.1.14.Программирование и отладка индивидуального задания
2.1.16.Программирование и отладка индивидуального задания
2.1.18.Программирование и отладка индивидуального задания

Задание №1

Написать программу, которая вычисляет среднюю (за неделю) температуру воздуха. Исходные данные должны вводиться во время работы программы. Рекомендуемый вид экрана программы приведен ниже (данные, введенные пользователем, выделены полужирным).

```
???????
??????????? ????
??? ?? ???????????????? -> 12??????? -> 10
????? -> 16??????? -> 18??????? -> 17??????? -> 16
??????????? -> 14
------------------------------??????? ??????????? ?? ??????:
 14.71 ????.
```

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

2.4 Текущий контроль (ТК) № 4

Тема занятия: 3.2.7.Арифметические команды. Режимы адресации

Метод и форма контроля: Домашняя работа (Опрос)

Вид контроля: Домашняя работа с аналитической направленностью

Дидактическая единица: 1.5 общую характеристику языков ассемблера; назначение, принципы построения и использования

Занятие(-я):

3.1.1.Память. Процессор. Регистры процессора. Система прерывания
3.1.2.Работа в разных системах счисления
3.1.3.Представление машинных команд
3.1.4.Программ DEBUG
3.2.1.Язык программирования Assembler. Структура команд. Описание данных
3.2.3.Структура .exe программ. Ассемблирование, компоновка и выполнение программ
3.2.5.Структура .com программ. Ассемблирование, компоновка и выполнение программ

Задание №1

Подготовьте реферат на тему "Язык программирования Ассемблер". по следующему плану:

1. ????????.2. ???????? ?????????????? ?? ????? ????????? - ??????? ?????????; - ??????? ?????????? ??? ???; - ????????? ??????????.3. ??????????? ????????? ? ????? ?????????. (?????????? ????? ????????????? ????? ????)4. ??????????.5. ?????? ??????????

Критерии оценки реферата

Изложенное понимание реферата как целостного авторского текста определяет критерии его оценки: степень раскрытия сущности вопроса; соблюдения требований к оформлению.

Степень раскрытия сущности вопроса:

- соответствие плана теме реферата;
- соответствие содержания теме и плану реферата;
- полнота и глубина знаний по теме;
- обоснованность способов и методов работы с материалом;
- умение обобщать, делать выводы, сопоставлять различные точки зрения по одному вопросу (проблеме).

Обоснованность выбора источников:

- оценка использованной литературы: привлечены ли наиболее известные работы по теме исследования (в т.ч. журнальные публикации последних лет, последние статистические данные, сводки, справки и т.д.).

Соблюдение требований к оформлению:

- насколько верно оформлены ссылки на используемую литературу, список литературы;
- оценка грамотности и культуры изложения (в т.ч. орфографической, пунктуационной, стилистической культуры), владение терминологией;
- соблюдение требований к объему реферата.

Оценка	*Показатели оценки*
5	Выполнены все требования к написанию и защите реферата: обозначена проблема и обоснована ее актуальность, сделан краткий анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объем, соблюдены требования к внешнему оформлению, даны правильные ответы на дополнительные вопросы.
4	Основные требования к реферату и его защите выполнены, но при этом допущены недочеты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объем реферата; имеются упущения в оформлении; на дополнительные вопросы при защите даны неполные ответы.
3	Имеются существенные отступления от требований к реферированию. В частности: тема освещена лишь частично; допущены фактические ошибки в содержании реферата или при ответе на дополнительные вопросы; во время защиты отсутствует вывод.

2.5 Текущий контроль (ТК) № 5

Тема занятия: 3.2.18.Программирование и отладка задания
Метод и форма контроля: Практическая работа (Опрос)
Вид контроля: Практическая работа с использованием технических средств
Дидактическая единица: 2.3 составлять и оформлять программы на языках программирования;
Занятие(-я):
2.1.20.Программирование и отладка индивидуального задания
3.2.2.Программирование и отладка задания
3.2.4.Программирование и отладка задания
3.2.6.Программирование и отладка задания
3.2.8.Программирование и отладка задания
3.2.10.Программирование и отладка задания
3.2.12.Программирование и отладка задания
3.2.14.Программирование и отладка задания
3.2.16.Программирование и отладка задания
Задание №1
Выполните задания *(представлен один из вариантов)***:**
Задание 1:

1. Занести номер дома в регистры AX, BX, CX, DX
2. Сложит содержимое регистров в DX
3. Очистить регистры AX, BX, CX

Задание 2:
Написать программу: найти сумму чисел.

1. Если сумма чисел положительное число, то число сдвинуть вправо, иначе – сдвинуть влево.

Задание 3:
Составить .exe программу:

1. Найти сумму первой строки задания
2. Найти сумму второй строки задания
3. Найти сумму третьей строки задания

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2
Q4R54)

Задание 4:
Составить .com программу:

1. Найти разность первой строки задания
2. Найти разность второй строки задания
3. Найти произведение третьей строки задания

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2
Q4R54)

Оценка	***Показатели оценки***
5	Выполнено 4 задания

4	Выполнено 3 задания, допущены ошибки
3	Выполнено 2 задания

Дидактическая единица: 2.4 тестировать и отлаживать программы;
Занятие(-я):
2.1.20.Программирование и отладка индивидуального задания
3.2.2.Программирование и отладка задания
3.2.4.Программирование и отладка задания
3.2.6.Программирование и отладка задания
3.2.8.Программирование и отладка задания
3.2.10.Программирование и отладка задания
3.2.12.Программирование и отладка задания
3.2.14.Программирование и отладка задания
3.2.16.Программирование и отладка задания
Задание №1
Выполните задания *(представлен один из вариантов)***:**
Задание 1:
Составить программу:

1. Найти сумму первой строки задания
2. Найти разность второй строки задания
3. Если сумма больше разности, то содержимое регистров поменять местами

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2

Задание 2:
Составить программу занесения в каждый байт буфера размером 25 байт числа - 0FCH.

Задание 3:
Составить программу, используя процедуру нахождения суммы чмсел:

1. Найти сумму первой строки задания
2. Найти сумму второй строки задания
3. Найти сумму третьей строки задания

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2
Q4R54)

Задание 4:

1. Составить программу, позволяющую преобразовать строку символов в числовой эквивалент и вычислить сумму чисел.
2. Символы ввести с клавиатуры.
3. Результат вывести на экран.

Q4R54)

Оценка	***Показатели оценки***
5	Выполнено 4 задания
4	Выполнено 3 задания, допущены ошибки
3	Выполнено 2 задания

3. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

№ семестра	Вид промежуточной аттестации
6	Дифференцированный зачет

Дифференцированный зачет может быть выставлен автоматически по результатам текущих контролей
Текущий контроль №1
Текущий контроль №2
Текущий контроль №3
Текущий контроль №4
Текущий контроль №5

Метод и форма контроля: Практическая работа (Опрос)
Вид контроля: По выбору выполнить два практических задания
Дидактическая единица для контроля:
1.1 общие принципы построения и использования языков программирования, их классификацию;
Задание №1 (из текущего контроля)
Заполните таблицу "Классификация языков программирования" недостающими данными:

Факторы	Характеристика	Группы	Примеры ЯП
Уровень ЯП	Степень близости ЯП к архитектуре компьютера	Низкий	
		Высокий	
		Сверхвысокий	Сетл
Специализация ЯП	Потенциальная или реальная область применения		Algol, Pascal, Basic Fortran, Cobol, Ada
		Процедурные	
		Непроцедурные	

Оценка	*Показатели оценки*

5	**Факторы**	**Характеристика**	**Группы**	**Примеры ЯП**
	Уровень ЯП	Степень близости ЯП к архитектуре компьютера	Низкий	***Ассемблер***
			Высокий	***Pascal, Basic, Fortran, C…***
			Сверхвысокий	Сетл
	Специализация ЯП	Потенциальная или реальная область применения	***Общего назначения***	Algol, Pascal, Basic
			Специализированные	Fortran, Cobol, Ada
	Алгоритмичность	***Возможность абстрагироваться от деталей алгоритма***	Процедурные	***Ассемблер, Pascal, Basic, Fortran***
			Непроцедурные	***Prolog***
4	Таблица заполнена не полностью (6 ячеек), допущены ошибки			
3	Таблица заполнена не полностью (4 ячейки), допущены ошибки			

Задание №2 (из текущего контроля)
Перечислите элементы языков программирования, и дайте определение каждому элементу.

Оценка	***Показатели оценки***
5	Перечислены следующие элементы, даны все определения: Алфавит - совокупность символов, отображаемых на устройствах печати и экрана и/или вводимых с клавиатуры терминала. Лексика - совокупность правил образования цепочек символов, образующих идентификаторы (переменные и метки), операторы, операции и другие лексические компанеты языка. Синтаксис - совокупность правил образования языковых конструкций, или предложения ЯП - блоков, процедур, составных операторов, опрераторов цикла, условных операторов и пр. Особенностью синтаксиса является принцип вложенности (рекурсивность)правил построения конструкций. Семантика - смысловое содержание конструкций, предложений языка, семантический анализ - это проверка смысловой правильности конструкции.

4	Перечислены все элементы, даны три определения. В определениях допущены неточности.
3	Перечислены все элементы, даны два определения. В определениях допущены неточности

Задание №3 (из текущего контроля)
Перечислите из каких частей состоит исходная прграмма, дайте определение каждой из частей.

Оценка	***Показатели оценки***
5	Перечислены все основные части, даны определения. 1. Раздел идентификации - раздел, содержащий наименование программы; 2. Раздел данных - идентификация переменных (объявление, писание); 3. Раздел процедур - программная часть, содержащая описание процессов обработки данных.
4	Перечислены все основные части, определения даны не все, или допущены неточности.
3	Перечислены не все основные части, в определениях допущены неточности.

Дидактическая единица для контроля:
1.2 современные интегрированные среды разработки программ;
Задание №1 (из текущего контроля)
Ответьте на следующие вопросы:
Вариант №1

1. После имени функции ставятся _____.
2. В чем особенность функции main ()?
3. Напишите оператор, выводящий на экран
    1. число 509;
    2. значение переменной x.
4. Напишите оператор, который получает с клавиатуры числовое значение и присваивает его переменной temp?
5. На какую величину увеличивает значение переменной операция инкримента?
6. Какой знак в языке C++ является разделителем и отмечает конец инструкции?

7. Каково назначение директивы # include в языке С++?
8. Как производится объявление константы в языке С++? Объявите константу Pi=3,14.
9. Каково назначение функции getch () в С++?
10. Каким образом осуществляется ввод/вывод данных?

Вариант №2

1. Тело функции заключается в _____.
2. Истинно ли следующее утверждение: в операции присваивания величина, стоящая слева от знака равенства, всегда равна величине, стоящей справа от знака равенства?
3. Запишите на языке С++ ab.
4. Какой заголовочный файл нужно включить в исходный текст, чтобы использовать объекты cin и count?
5. Напишите оператор, увеличивающий значение переменной temp на 23 с одновременным присваиванием. Напишите аналогичный оператор, не использующий сложения с присваиванием.
6. Какие значения выведут на экран два указанных оператора, если начальное значение переменной w равно 20?

    1. Cout << w --;
    2. Cout << ++ w;
7. С какой целью заголовочные файлы включаются в тексты программ?
8. Перечислите основные типы данных в С++?
9. Каково назначение объектов cin / cout?
10. Как производится объявление переменной в языке С++?

Оценка	***Показатели оценки***
3	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 5-6
4	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 7-8
5	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 9-10

Дидактическая единица для контроля:

1.3 процесс создания программ;

Задание №1 (из текущего контроля)

1. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << a << "=Z(" << b << ")";
```

Здесь a и b — целые переменные.

2. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << "a,=Z(" << b << ")";
```

Здесь a и b — целые переменные.

3. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << "Z(a)=(b)";
```

Здесь a и b — целые переменные.

4. Что будет выведено на экран в результате выполнения фрагмента программы

```
a=5; b=3;
cout << "Z(" << a << ")=(" << b << ")";
```

Здесь a и b — целые переменные.

5. Запишите оператор для вывода значений целых переменных a=5 и b=3 в потоксout в следующем формате:

```
5+3=?
```

6. Запишите оператор для вывода значений целых переменных a=5 и b=3 в поток cout в следующем формате:

```
Z(5)=F(3)
```

7. Запишите оператор для вывода значений целых переменных a=5 и b=3 в поток cout в следующем формате:

```
a=5;b=3;
```

8. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 5 ) a = 5;
```

9. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a > 5 ) a = 12;
```

10. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 5 )
a = a + 12;
else
a = a - 7;
```

11. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 15 )
a = a + 12;
else
a = a - 7;
```

12. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a < 5 ) a = a + 12;
else a = a - 7;
if ( a < 5 ) a = a + 12;
else a = a - 7;
```

13. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10;
if ( a > 5 ) a = a + 12;
else a = a - 7;
if ( a > 5 ) a = a + 12;
else a = a - 7;
```

14. Определите значение переменной «a» после выполнения фрагмента программы:

```
a = 10; b = 3;
if ( a > b ) b = b + 12;
else a = a - 7;
if ( a > b ) a = a + 12;
else a = a - 7;
```

Оценка	***Показатели оценки***

3	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 6-8
4	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 9-12
5	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 13-14

Дидактическая единица для контроля:

1.4 стандарты языков программирования;

Задание №1 (из текущего контроля)

1. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 2 раза?

```
i = 4;
while ( i <= ... ) {
cout << "Привет!\n";
i++;
}
```

2. Сколько раз будет выполнен этот цикл?

```
i = 6;
while ( i < 5 ) {
cout << "Привет!\n";
i++;
}
```

3. Какое число будет выведено на экран в результате выполнения этого цикла?

```
i = 5;
cout << "9";
while ( i < 5 ) {
cout << i;
i++;
}
```

4. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 4 раза?

```
i = 3;
while ( i >= ... ) {
cout << "Привет!\n";
i--;
}
```

5. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 2 раза?

```
i = ...;
while ( i >= 2 ) {
cout << "Привет!\n";
i--;
}
```

6. Какое число нужно написать вместо многоточия, чтобы цикл выполнился ровно 2 раза?

```
i = ...;
while ( i <= 2 ) {
cout << "Привет!\n";
i++;
}
```

7. Чему будет равно значение переменной «a» после выполнения этого цикла?

```
i = 4;
a = 12;
while ( i < 5 ) {
a += i;
i++;
}
```

8. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 10;
for( i=1; i<=2; i++ )
a -= 1;
```

9. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 10;
for ( i=0; i<=3; i++ )
```

```
a -= i;
```

10. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 2;
for ( i=0; i<=3; i++ )
a += i;
```

11. Чему будет равно значение целой переменной «a» после выполнения этого фрагмента программы?

```
a = 4;
for ( i=1; i<=4; i++ )
a += a;
```

12. Какое число будет выведено на экран после выполнения этого фрагмента программы?

```
for ( i=2; i>=0; i-- )
cout << i;
```

13. Какое число будет выведено на экран после выполнения этого фрагмента программы?

```
cout << 4;
for ( i=4; i>=3; i-- )
cout << 2*i;
```

14. Какое число будет выведено на экран после выполнения этого фрагмента программы?

```
cout << 1;
for ( i=2; i<=0; i++ )
cout << i;
```

Оценка	***Показатели оценки***
3	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 6-8
4	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 9-12

5	За правильный ответ на вопросы или верное решение задачи выставляется положительная оценка – 1 балл. Количество набранных баллов 13-14

Дидактическая единица для контроля:

1.5 общую характеристику языков ассемблера; назначение, принципы построения и использования

Задание №1 (из текущего контроля)

Подготовьте реферат на тему "Язык программирования Ассемблер". по следующему плану:

1. ????????.2. ???????? ?????????????? ?? ????? ????????
- ??????? ????????; - ??????? ?????????? ??? ???; - ???????? ??????????.3. ??????????? ????????? ????? ????????. (?????????? ????? ????????????? ????? ????)4. ??????????.5. ?????? ??????????

Критерии оценки реферата

Изложенное понимание реферата как целостного авторского текста определяет критерии его оценки: степень раскрытия сущности вопроса; соблюдения требований к оформлению.

Степень раскрытия сущности вопроса:

- соответствие плана теме реферата;
- соответствие содержания теме и плану реферата;
- полнота и глубина знаний по теме;
- обоснованность способов и методов работы с материалом;
- умение обобщать, делать выводы, сопоставлять различные точки зрения по одному вопросу (проблеме).

Обоснованность выбора источников:

- оценка использованной литературы: привлечены ли наиболее известные работы по теме исследования (в т.ч. журнальные публикации последних лет, последние статистические данные, сводки, справки и т.д.).

Соблюдение требований к оформлению:

- насколько верно оформлены ссылки на используемую литературу, список литературы;
- оценка грамотности и культуры изложения (в т.ч. орфографической, пунктуационной, стилистической культуры), владение терминологией;
- соблюдение требований к объему реферата.

Оценка	***Показатели оценки***

5	Выполнены все требования к написанию и защите реферата: обозначена проблема и обоснована ее актуальность, сделан краткий анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объем, соблюдены требования к внешнему оформлению, даны правильные ответы на дополнительные вопросы.
4	Основные требования к реферату и его защите выполнены, но при этом допущены недочеты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объем реферата; имеются упущения в оформлении; на дополнительные вопросы при защите даны неполные ответы.
3	Имеются существенные отступления от требований к реферированию. В частности: тема освещена лишь частично; допущены фактические ошибки в содержании реферата или при ответе на дополнительные вопросы; во время защиты отсутствует вывод.

Дидактическая единица для контроля:

1.6 базовые структуры данных, их преимущества и недостатки, алгоритмы работы с базовыми структурами данных;

Задание №1 (из текущего контроля)

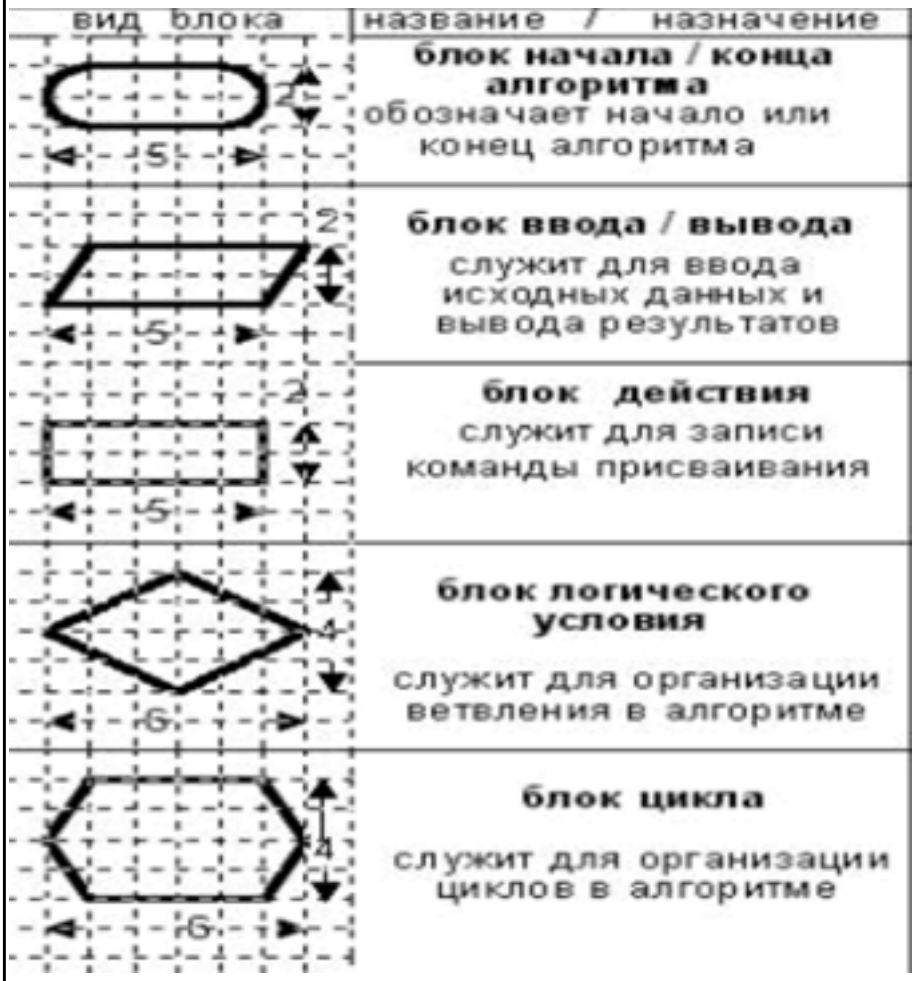
Дайте определение понятию "Алгоритм";

Перечислите и объясните свойства алгоритма;

Назовите формы записи алгоритмов;

Назовите и изобразите 5 основных элементов блок-схем.

Оценка	***Показатели оценки***

5	Алгоритм - система четких однозначных указаний, которая определяет последовательность действий над некоторыми объектами и после конечного числа шагов приводит к плучению требуемого результата. Свойства алгоритмов: 1. Дискретность - разделение выполнения решения задачи на отдельные операции. 2. Определенность - каждая команда определяет однозначное действие. 3. Понятность - система команд понятна исполнителю. 4. Результативность - исполнение алгоритма должно закончиться за конечное число шагов. Формы записи алгоритма: - словесная; - графическая; - псевдокоды (алгоритмический язык); - программная. Основные элементы блок-схем: 
4	На один из вопросов не дан ответ / допущены ошибки в формулировке ответа

3	На два вопроса не даны ответы / на один вопрос не дан ответ и допущены ошибки в формулировке ответа

Дидактическая единица для контроля:

2.1 формализовать поставленную задачу;

Задание №1 (из текущего контроля)

Построить алгоритм решения задач в виде блок-схем:

1. Дано значение температуры T в градусах Фаренгейта. Определить значение этой же температуры в градусах Цельсия. Температура по Цельсию TC и температура по Фаренгейту TF связаны следующим соотношением: TC = (TF − 32)·5/9.
2. Известно, что X кг конфет стоит A рублей. Определить, сколько стоит 1 кг и Y кг этих же конфет.
3. Поменять местами содержимое переменных A и B и вывести новые значения A и B.
4. Для данного вещественного x найти значение следующей функции f, принимающей вещественные значения:

$$f(x) = \begin{cases} 2\cdot\sin(x), & \text{если } x > 0, \\ 6 - x, & \text{если } x \le 0. \end{cases}$$

5. Даны целые числа K и N ($N > 0$). Вывести N раз число K.

Оценка	***Показатели оценки***
5	Алгоритм построен верно для всех задач
4	Алгоритм не построен для одной задачи / Допущены ошибки в построении блок-схемы.
3	Алгоритм не построен для двух задач / Алгоритм не построен для одной задачи и допущены ошибки в построении блок-схем.

Дидактическая единица для контроля:

2.2 применять полученные знания к различным предметным областям;

Задание №1 (из текущего контроля)

Написать программу, которая вычисляет среднюю (за неделю) температуру воздуха. Исходные данные должны вводиться во время работы программы. Рекомендуемый вид экрана программы приведен ниже (данные, введенные пользователем, выделены полужирным).

???????
??????????? ????

??? ?? ??????????????? -> **12**??????? -> **10**
????? -> **16**??????? -> **18**??????? -> **17**??????? -> **16**
?????????? -> **14**
----------??????? ?????????? ?? ??????: 14.71 ????.

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Дидактическая единица для контроля:
2.3 составлять и оформлять программы на языках программирования;
Задание №1 (из текущего контроля)
Написать программу, которая выводит на экран фразу "Каждый охотник желает знать, где сидят фазаны", позволяющую запомнить порядок следования цветов радуги (первая буква слова кодирует цвет: каждый — красный, охотник — оранжевый, желает — желтый, знать — зеленый, где — голубой, сидят — синий, фазаны — фиолетовый). Каждое слово фразы должно быть выведено наиболее подходящим цветом.

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2 (из текущего контроля)
Выполните задания *(представлен один из вариантов)***:**
Задание 1:

1. Занести номер дома в регистры AX, BX, CX, DX
2. Сложит содержимое регистров в DX
3. Очистить регистры AX, BX, CX

Задание 2:
Написать программу: найти сумму чисел.

1. Если сумма чисел положительное число, то число сдвинуть вправо, иначе – сдвинуть влево.

Задание 3:
Составить .exe программу:

1. Найти сумму первой строки задания
2. Найти сумму второй строки задания
3. Найти сумму третьей строки задания

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2
Q4R54)

Задание 4:
Составить .com программу:

1. Найти разность первой строки задания
2. Найти разность второй строки задания
3. Найти произведение третьей строки задания

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2
Q4R54)

Оценка	***Показатели оценки***
5	Выполнено 4 задания
4	Выполнено 3 задания, допущены ошибки
3	Выполнено 2 задания

Дидактическая единица для контроля:
2.4 тестировать и отлаживать программы;
Задание №1 (из текущего контроля)
Написать *функцию*, которая вычисляет сопротивление цепи, состоящей из двух резисторов, которые могут быть соединены последовательно или параллельно.

Функция должна проверять корректность параметров: если неверно указан тип соединения, то функция должна возвращать –1.

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2 (из текущего контроля)
Написать программу, которая дописывает в находящийся на диске компьютера файл contacts.txt имя, фамилию и номер телефона, например, вашего товарища. Если файла на диске нет, то программа должна создать его. В файле каждый элемент данных (имя, фамилия, телефон) должен находиться в отдельной строке. Ниже приведен рекомендуемый вид экрана во время работы программы.

```
????
?????? ?
????????? ? ????
?????? ???????????????? -> ?????????? -> ????
??????? -> 234-84-37
?????????? ??????????? ?????????? ??????? <Enter>
```

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №3 (из текущего контроля)
Выполните задания *(представлен один из вариантов)***:**
Задание 1:
Составить программу:

1. Найти сумму первой строки задания
2. Найти разность второй строки задания
3. Если сумма больше разности, то содержимое регистров поменять местами

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2

Задание 2:
Составить программу занесения в каждый байт буфера размером 25 байт числа - 0FCH.

Задание 3:
Составить программу, используя процедуру нахождения суммы чмсел:

1. Найти сумму первой строки задания
2. Найти сумму второй строки задания
3. Найти сумму третьей строки задания

-45,817,-58,33,177
E1 A5 A3 AC A5 AD E2
Q4R54)

Задание 4:

1. Составить программу, позволяющую преобразовать строку символов в числовой эквивалент и вычислить сумму чисел.
2. Символы ввести с клавиатуры.
3. Результат вывести на экран.

Q4R54)

Оценка	***Показатели оценки***
5	Выполнено 4 задания
4	Выполнено 3 задания, допущены ошибки
3	Выполнено 2 задания

Дидактическая единица для контроля:
2.5 формализировать прикладные задачи и интерпретировать их в терминах программирования;
Задание №1 (из текущего контроля)

Написать программу, которая вычисляет величину тока, потребляемого электроприбором (I = P/U, где: I — ток, А; P — мощность, Вт; U — напряжение, В). Программа должна проверять правильность введенных пользователем данных и, если они неверные (делитель равен нулю), выводить сообщение об ошибке. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
??? ? ????????????
???????????, ?? -> 60??????????, ? -> 0
??????! ?????????? ?? ?????? ???? ????? ????.
```

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2 (из текущего контроля)
Написать программу, которая преобразует введенное пользователем десятичное число в двоичное. Рекомендуемый вид экрана программы приведен ниже.

```
????????????? ?????????? ????? ? ????????????? ????? ???
?? ?? 0 ?? 255 ? ??????? <
Enter>->49
?????????? ????? 49 ???????????? ???????? 0011000??? ????
?????? ??????? <Enter>
```

Оценка	*Показатели оценки*
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора