

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)**

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

дисциплины:

направление подготовки:

направленность:

форма обучения:

Программирование

21.03.01 Нефтегазовое дело

**Эксплуатация и обслуживание объектов
добычи нефти**

очно-заочная

Фонд оценочных средств разработан в соответствии с утвержденным учебным планом от 22.04.2019 г. и требованиями ОПОП ВО по направлению подготовки 21.03.01 Нефтегазовое дело, направленность **Эксплуатация и обслуживание объектов добычи нефти** к результатам освоения дисциплины «Программирование».

Фонд оценочных средств рассмотрен на заседании кафедры Транспорта и технологий нефтегазового комплекса

Протокол № 9 от «15» мая 2019 г.

Заведующий кафедрой  А.В. Козлов

СОГЛАСОВАНО:

Заведующий выпускающей кафедрой  А.В. Козлов

«15» мая 2019 г.

Фонд оценочных средств разработал:
Лаптева С.В., доцент кафедры ТТНК, к.п.н.



1. Результаты обучения по дисциплине

Таблица 1.1

Код и наименование компетенции	Код и наименование индикатора достижения компетенции (ИДК)	Код и наименование результата обучения по дисциплине (модулю)
<p>УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач</p>	<p>УК-1.1. Выбор информационных ресурсов для поиска информации в соответствии с поставленной задачей</p>	<p>Знать (З1): методы выбора информационных ресурсов для поиска информации; основы информационной и библиографической культуры; особенности поиска информации в глобальной сети Интернет</p>
		<p>Уметь (У1): осуществлять выбор информационных ресурсов для решения поставленной задачи; использовать системный подход к построению алгоритма решения задачи; анализировать найденную информацию для дальнейшего использования в решении поставленной задачи</p>
		<p>Владеть (В1): навыками поиска информации для решения поставленной задачи; навыками анализа собранной информации для решения поставленной задачи; навыками поиска информации в глобальной сети Интернет</p>
	<p>УК-1.2. Оценка соответствия выбранного информационного ресурса критериям полноты и аутентичности</p>	<p>Знать (З2): технологию оценки соответствия выбранного информационного ресурса критериям полноты и аутентичности при решении профессиональных задач</p>
		<p>Уметь (У2): оценивать выбранный информационный ресурс согласно критериям; оценивать выбранное программное обеспечение для решения профессиональных задач</p>
		<p>Владеть (В2): навыками оценки соответствия выбранного информационного ресурса критериям полноты и аутентичности при решении профессиональных задач</p>
<p>УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений</p>	<p>УК-2.5. Составление последовательности (алгоритма) решения задачи</p>	<p>Знать (З3): понятие «алгоритма»; свойства алгоритма; этапы решения задачи; правила построения блок-схем; основные алгоритмические структуры (следование, ветвление, цикл) и особенности их построения</p>
		<p>Уметь (У3): распознавать тип алгоритмической структуры задачи; разрабатывать алгоритм решения задачи согласно заданному условию; «читать» блок-схемы</p>
		<p>Владеть (В3): навыками построения блок-схем согласно условию вычислительной задачи; навыками разработки алгоритма решения задачи согласно заданному условию</p>
<p>ОПК 2. Способен участвовать в проектировании технических объектов, систем и технологических процессов с учетом экономических, экологических, социальных и других ограничений</p>	<p>ОПК-2.8. Применение навыков работы с ЭВМ, используя новые методы и пакеты программ</p>	<p>Знать (З4): основные принципы функционирования компьютерной техники и ее применения для решения задач своей профессиональной деятельности; принципы работы с прикладным программным обеспечением; основы компьютерной грамотности; основные методы и способы решения поставленных задач с применением ЭВМ и прикладного программного обеспечения; требования к обеспечению информационной безопасности</p>
		<p>Уметь (У4): применять основные принципы функционирования компьютерной техники и принципы работы с прикладным программным обеспечением при решении поставленных задач; применять основные методы и способы решения поставленных задач при использовании ЭВМ и прикладного программного обеспечения; использовать требования к обеспечению информационной безопасности при работе на ЭВМ и в глобальных компьютерных сетях</p>
		<p>Владеть (В4): навыками работы с прикладным программным обеспечением при решении поставленных задач; навыками применения методов и способов решения</p>

		задач при использовании ЭВМ и прикладного программного обеспечения; навыками обеспечения информационной безопасности при работе на ЭВМ и в глобальных компьютерных сетях
<p>ОПК 4. Способен проводить измерения и наблюдения, обрабатывать и представлять экспериментальные данные</p>	<p>ОПК-4.5. Проведение эксперимента с использованием пакетов программ</p>	<p>Знать (З5): особенности и алгоритм проведения экспериментов; технологию проведения исследований и различных этапов проектирования экспериментальных работ; принципы работы в конкретных прикладных программных подуктах</p>
		<p>Уметь (У5): использовать различные средства обработки информации; строить алгоритм проведения экспериментальных работ; использовать возможности прикладных программных продуктов для решения профессиональных задач, связанных с проектированием</p>
		<p>Владеть (В5): навыками планирования и организации различных исследований и разработок, проведения экспериментальных работ; навыками применения прикладных программных продуктов для решения профессиональных задач, связанных с проектированием</p>
<p>ОПК 5. Способен решать задачи в области профессиональной деятельности с применением современных информационных технологий и прикладных аппаратно-программных средств</p>	<p>ОПК-5.5. Применение прикладного программного обеспечения для проведения инженерных расчетов</p>	<p>Знать (З6): назначение и возможности прикладного программного обеспечения конкретного направления; принципы работы в конкретной программной системе</p>
		<p>Уметь (У6): осуществлять выбор прикладных программных продуктов среди подобных систем для наиболее эффективного использования в решении профессиональных задач; использовать прикладное программное обеспечение в решении профессиональных задач</p>
		<p>Владеть (В6): навыками выбора прикладного программного обеспечения; навыками применения прикладного программного обеспечения в осуществлении инженерных расчетов</p>
<p>ОПК 6. Способен принимать обоснованные технические решения в профессиональной деятельности, выбирать эффективные и безопасные технические средства, и технологии.</p>	<p>ОПК-6.7. Решение стандартных задач профессиональной деятельности на основе современных информационных технологий и с учетом требований информационной безопасности</p>	<p>Знать (З7): основы компьютерной грамотности; правовые нормы по защите информации и соблюдению прав интеллектуальной собственности; требования к обеспечению информационной безопасности</p>
		<p>Уметь (У7): соблюдать основные требования информационной безопасности при решении профессиональных задач</p>
		<p>Владеть (В7): навыками анализа и выбора методов и средств обеспечения информационной безопасности; навыками применения средств информационной безопасности в решении профессиональных задач</p>

2. Формы аттестации по дисциплине

2.1. Форма промежуточной аттестации: экзамен.

Способ проведения промежуточной аттестации: **письменный экзамен.**

2.2. Формы текущей аттестации:

Таблица 2.1

№ п/п	Форма обучения
	ОЗФО
1	Теоретический коллоквиум
2	Теоретический коллоквиум; выполнение лабораторной работы
3	Тест
4	Тест; выполнение лабораторной работы
5	Тест; выполнение лабораторной работы; сообщение (реферат)

3. Результаты обучения по дисциплине, подлежащие проверке при проведении текущей и промежуточной аттестации

Таблица 3.1

№ п/п	Структурные элементы дисциплины/модуля		Код результата обучения по дисциплине/модулю	Оценочные средства	
	Номер раздела	Дидактические единицы (предметные темы)		Текущая аттестация	Промежуточная аттестация
1	1	Этапы проектирования и жизненный цикл программных продуктов	31, У1, В1, 32, У2, В2, 33, У3, В3, 34, У4, В4, 37, У7, В7	Теоретический коллоквиум	Экзаменационные вопросы и задания
2	2	Программирование основных алгоритмических структур на языке программирования	31, У1, В1, 32, У2, В2, 33, У3, В3, 34, У4, В4, 35, У5, В5, 36, У6, В6, 37, У7, В7	Теоретический коллоквиум; выполнение лабораторной работы	Экзаменационные вопросы и задания
3	3	Структурный подход к построению программ	31, У1, В1, 32, У2, В2, 33, У3, В3, 34, У4, В4, 35, У5, В5, 36, У6, В6, 37, У7, В7	Тест	Экзаменационные вопросы и задания
4	4	Типы данных в языке программирования	31, У1, В1, 32, У2, В2, 33, У3, В3, 34, У4, В4, 35, У5, В5, 36, У6, В6, 37, У7, В7	Тест; выполнение лабораторной работы	Экзаменационные вопросы и задания
5	5	Типы данных, определяемые пользователем	31, У1, В1, 32, У2, В2, 33, У3, В3, 34, У4, В4, 35, У5, В5, 36, У6, В6, 37, У7, В7	Тест; выполнение лабораторной работы; сообщение (реферат)	Экзаменационные вопросы и задания

4. Фонд оценочных средств

4.1. Фонд оценочных средств, позволяющие оценить результаты обучения по дисциплине, включает в себя оценочные средства для текущей аттестации и промежуточной аттестации.

4.2. Фонд оценочных средств для **текущей** аттестации включает:

- комплект вопросов для теоретического коллоквиума № 1 по теме «Этапы проектирования и жизненный цикл программных продуктов» – 8 вопросов (Приложение 1);
- шаблоны отчетов по лабораторным работам (№№ 1–5) – 5 шт. (приведены в методических указаниях по выполнению лабораторных работ);

- комплект вопросов для теоретического коллоквиума № 2 по теме «Программирование основных алгоритмических структур на языке программирования» – 14 вопросов (Приложение 2);
- комплект тестовых заданий № 1 по теме «Структурный подход к построению программ» – 20 шт. (Приложение 3);
- комплект тестовых заданий № 2 по теме «Типы данных в языке программирования» – 25 шт. (Приложение 4);
- комплект тестовых заданий № 3 по теме «Типы данных, определяемые пользователем» – 25 шт. (Приложение 5);
- комплект тем сообщений (рефератов) – 25 шт. (Приложение 6).

4.3. Фонд оценочных средств для **промежуточной** аттестации включает:

- комплект вопросов к экзамену – 30 шт. (Приложение 7).

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)**

Кафедра Транспорта и технологий нефтегазового комплекса

Перечень вопросов к теоретическому коллоквиуму № 1

Форма проведения - *письменный* коллоквиум.

1. Необходимость постановки задачи.
2. Понятие жизненного цикла программного обеспечения.
3. Процессы ЖЦ ПО.
4. Модели ЖЦ ПО.
5. Сравнительная характеристика моделей ЖЦ ПО.
6. Этапы разработки ЖЦ ПО.
7. Виды тестирования ПО.
8. Адекватность модели.

Критерии оценки

При оценке знаний обучающиеся получают 8 вопросов и отвечают письменно.

№ задания	Задание выполнено полностью	Задание выполнено, но имеют недочеты	Задание не выполнено
За каждое задание	0,5	0,2–0,4	0–0,1
Максимальное количество баллов – 4 балла			

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
 (ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
 ВЫСШЕГО ОБРАЗОВАНИЯ
«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
 (Филиал ТИУ в г.Ноябрьске)

Кафедра Транспорта и технологий нефтегазового комплекса

Комплект вопросов для теоретического коллоквиума № 2

Форма проведения - *письменный* коллоквиум.

1. Понятие лексемы.
2. Алфавит языка.
3. Основные типы данных.
4. Использование ключевого слова `unsigned` при определении (объявлении или инициализации) переменной.
5. Отличие друг от друга понятий «определение», «объявление» и «инициализация» переменных.
6. Основные арифметические операции.
7. Основные логические операции.
8. Пример операции присваивания.
9. Назначение символов `{ }`.
10. Оператор ветвления. Его запись для стандартного ветвления.
11. Оператор ветвления. Его запись для сокращенного ветвления.
12. Оператор цикла с параметром. Его запись в программе.
13. Оператор цикла с предусловием. Его запись в программе.
14. Оператор цикла с постусловием. Его запись в программе.

Критерии оценки

При оценке знаний обучающиеся получают 10 вопросов и отвечают письменно.

№ задания	Задание выполнено полностью	Задание выполнено, но имеют недочеты	Задание не выполнено
За каждое задание	0,5	0,2–0,4	0–0,1
Максимальное количество баллов – 7 баллов			

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ

«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)

Кафедра Транспорта и технологий нефтегазового комплекса

Комплект тестовых вопросов к тесту № 1

1. Какую функцию должны содержать все программы на C++?

- a) main()
- б) system()
- program()
- start()

2. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется:

- интерпретатор
- сканер
- компилятор
- транслятор

3. Цикл с постусловием?

- while
- do while
- for

4. Какими знаками заканчивается большинство строк кода в Си++?

- ; (точка с запятой)
- , (запятая)
- : (двоеточие)
- . (точка)

5. Какому зарезервированному слову программа передаёт управление в случае, если значение переменной или выражения оператора switch не совпадает ни с одним константным выражением?

- default
- all
- contingency
- other

6. Укажите правильное определение функции `main` в соответствии со спецификацией стандарта ANSI

- void main(void)
- void main()
- int main(void)
- int main()

7. Чему будет равна переменная a, после выполнения этого кода `int a; for(a = 0; a < 10; a++) {}`?

1
10
9

8. Какое значение, по умолчанию, возвращает программа операционной системе в случае успешного завершения?

-1
1
0
Программа не возвращает значение.

9. Какой оператор не допускает перехода от одного константного выражения к другому?

`break;`
`Stop;`
`end;`
точка с запятой

10. Чтобы подключить заголовочный файл в программу на C++, например `iostream` необходимо написать:

```
include <iostream,h>  
#include <> с iostream внутри скобок  
#include <>; с iostream.h внутри скобок  
include (iostreamh)
```

11. Что будет напечатано?

```
int main()  
{  
    for (int i = 0; i < 4; ++i)  
    {  
        switch (i)  
        {  
            case 0 : std::cout << "0";  
            case 1 : std::cout << "1"; continue;  
            case 2 : std::cout << "2"; break;  
            default : std::cout << "D"; break;  
        }  
        std::cout << ".";  
    }  
    return 0;  
}
```

0.1.2.

01.2.D.

011.2.D

Ошибка компиляции в строке 10

0112.D.

12. Укажите объектно-ориентированный язык программирования

Все варианты ответов

C++

Eiffel

Java

13. Выберите правильный вариант объявления константной переменной в C++, где `type` - тип данных в C++ `variable` - имя переменной `value` - константное значение

```
const variable = value;  
const type variable = value;  
const type variable := value;
```

14. Каков результат работы следующего фрагмента кода?

```
int x = 0;  
switch(x)  
{  
  case 1: cout << "Один";  
  case 0: cout << "Нуль";  
  case 2: cout << "Привет мир";  
}
```

Привет мир

НульПривет мир

Один

Нуль

15. В приведённом коде измените или добавьте один символ, чтобы код напечатал 20 звёздочек (*).

```
int i, N = 20;  
for(i = 0; i < N; i--)  
  printf("*");
```

a) `int i, N = 20;`
`for(i = 19; i < N; i--)`
 `printf("*");`

б) `int i, N = 20;`
`for(i = 20; i < N; i--)`
 `printf("*");`

в) `int i, N = 20;`
`for(i = 0; i < N; N--)`
 `printf("*"`

г) `int i, N = 40;`
`for(i = 0; i < N; i--)`
 `printf("*");`

16. Укажите правильное приведение типа данных:

a `(char)`;

`(char) a`;

`char: a`;

to `(char, a)`;

17. Какой из следующих операторов - оператор сравнения двух переменных?

==

:=

equal

=

18. Какой из ниже перечисленных операторов, не является циклом в C++?

do while

repeat until

for

while

19. До каких пор будут выполняться операторы в теле цикла `while (x < 100)`?

Пока x меньше или равен ста

Пока x строго меньше ста

Пока x равен ста

Пока x больше ста

20. Цикл с предусловием?

while

do while

for

Общая сумма баллов – 8 баллов.

За каждый правильный ответ – 0,4 балла.

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ

«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)

Кафедра Транспорта и технологий нефтегазового комплекса

Комплект тестовых вопросов к тесту № 2

1. Что такое функция?
 - a) Некоторая часть программы, содержащая описание переменных и констант основной программы
 - b) Некоторая часть программы, имеющая собственное имя и которая может вызываться из основной программы**
 - c) Некоторая часть программы, содержащая вредоносный код, и блокирует определенные действия системы
 - d) Некоторая часть программы, в которой происходит начальная инициализация всех полей структур, массивов, переменных.

2. Что такое массив?
 - a) Именованный набор переменных имеющих различные типы данных, и располагающихся в одной памяти
 - b) Именованный набор переменных и функций, которые располагаются в одной области памяти
 - c) Именованный набор переменных имеющих один тип данных, и располагающихся в одной области памяти**
 - d) Именованный набор переменных имеющих символьный тип данных, и располагающихся в одной области памяти

3. Как написать следующее выражение на языке C «Переменной a присвоено значение b»?
 - a) `a==b`
 - b) `a=b`**
 - c) `b=a`
 - d) `a:=b`

4. Как написать следующее выражение «Второму элементу массива Myarray присвоено значение пяти»?
 - a) `int [1] Myarray=«пять»`
 - b) `int Myarray [1] = 5`**
 - c) `int Myarray [2] = «пять»`
 - d) `int Myarray [2] = 5`

5. Как написать следующее выражение «Если переменная index больше size то мы инкрементируем переменную count»?
 - a) `if (index>size) { count++; }`**

- b) `if (index<size) { count--; }`
 c) `if (index>=size) { ++count; }`
 d) `if (index<size) { --count; }`
6. Какой диапазон значений имеет тип `int` для 32-разрядных вычислительных систем:
 a) от 0 до 255
 b) от -32768 до 32767
 c) от 0 до 65535
 d) **от 0 до 4 294 967 295**
7. Какой размер в байтах имеет переменная вещественного типа `float`
 a) 2
 b) **4**
 c) 8
 d) 10
8. Дан массив `int L[3][3] = { { 2, 3, 4 }, { 3, 4, 8 }, { 1, 0, 9 } }`; Чему будет равно значение элемента этого массива `L[1][2]`
 a) 2
 b) 3
 c) 4
 d) **8**
9. Объявление `char *buf;` соответствует
 a) созданию символьной переменной `buf`
 b) созданию строковой переменной `buf`
 c) **созданию указателя `buf` на символьное значение**
 d) созданию указателя `buf` на строку
10. Что называется прототипом функции?
 a) описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров
 b) описание функции, включая ее имя, тип возвращаемого значения, типы параметров
 c) **имя функции и тип возвращаемого значения**
 d) описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров, тело функции
11. Как обозначается в языке C (C++) следующий режим работы с потоком - создание нового файла для записи и чтения?
 a) `a+`
 b) `wb`
 c) **`w+`**
 d) `w+b`
12. Какая функция, описанная в заголовочном файле читает строку символов из файла?
 a) `gets()`
 b) `fputs()`
 c) **`fgets()`**
 d) `fscanf()`
13. Какой размер массива `M` будет после выполнения кода:
`char M[]="\nGoodlive" ?`
 a) 10

- b) 8
 - c) **9**
 - d) Не определен
14. В каких случаях необходимо использовать оператор return в теле функции?
- a) Всегда
 - b) **если необходимо, чтобы функция вернула значение**
 - c) если необходимо обеспечить выход из функции в произвольном месте
 - d) если указан тип возвращаемого значения, в том числе и void
15. При открытии файла выполняется следующее действие:
- a) физический файл связывается с логическим (файловой переменной)
 - b) устанавливается тип файла (текстовый или бинарный)
 - c) **устанавливается вид (режим) использования файла**
 - d) функцией открытия файла возвращается результат (ошибка)
16. Какое ключевое слово языка C++ используется для описания структурированного типа данных, все элементы которого в памяти начинаются с одного байта?
- a) struct
 - b) **union**
 - c) enum
 - d) template
17. Каким способом можно задать многострочный комментарий в языке C++
- a) **/*комментарии к программе*/**
 - b) //комментарии к программе//
 - c) //комментарии к программе
 - d) {комментарии к программе}
18. Логическое выражение может возвращать результат типа
- a) integer
 - b) **boolean**
 - c) char
 - d) logical
19. Выберите правильный вариант записи на языке C формулы $0 < x < 10$
- a) $x > 0, x \leq 10$
 - b) $0 < x \leq 10$
 - c) $x > 0 \text{ AND } x \leq 10$
 - d) **$(x > 0) \text{ AND } (x < 10)$**
20. Укажите правильный вариант записи условного оператора в языке C
- a) IF $x > 0$ Do $y := \text{sqrt}(x)$
 - b) IF $y := \text{sqrt}(x)$ then $x > 0$
 - c) IF $x > 0$ then $y := \text{sqrt}(x)$
 - d) **IF $(x > 0)$ { $y := \text{sqrt}(x)$ }**
21. Выберите правильный вариант записи на языке C следующего условия: « x принадлежит диапазону $[0; 10)$ »
- a) $x \geq 0; x < 10$
 - b) $0 \leq x < 10$
 - c) $(x > 0 \text{ AND } (x \leq 10))$

d) $(x \geq 0) \text{ AND } (x < 10)$

22. Укажите группу, содержащую последовательность правильно записанных на языке C знаков операций отношений

- a) $\sim >, <, =, ?$
- b) $=, <>, ><, >$
- c) $=, \geq, \leq, !=$
- d) $\sim \Rightarrow, = <, =, <$

23. Тело какого цикла всегда будет выполнено хотя бы один раз, независимо от истинности условия:

- a) While
- b) **Do While**
- c) For
- d) Нет такого цикла в языке C

24. В результате выполнения кода

```
int i=2;  switch (i)  { case 1: i += 2; case 2: i *= 3; case 6: i /= 2;
           default:  ;  }
```

- a) переменная i примет значение 6
- b) **переменная i примет значение 3**
- c) переменная i примет значение 2
- d) тело оператора switch не поменяет значение переменной i

25. Укажите директиву препроцессора, которую необходимо подключить для организации форматированного ввода-вывода данных:

- a) `#include <iostream.h>`
- b) **`#include <stdio.h>`**
- c) `#include <math.h>`
- d) `#include <conio.h>`

Общая сумма баллов – 10 баллов.

За каждый правильный ответ – 0,4 балла.

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ

«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)

Кафедра Транспорта и технологий нефтегазового комплекса

Комплект тестовых вопросов к тесту № 3

1. Класс - это:

- любой тип данных, определяемый пользователем
- тип данных, определяемый пользователем и сочетающий в себе данные и функции их обработки; +
- структура, для которой в программе имеются функции работы с нею

2. Членами класса могут быть

- как переменные, так и функции, могут быть объявлены как `private` и как `public`; +
- только переменные, объявленные как `private`
- только функции, объявленные как `private`
- только переменные и функции, объявленные как `private`
- только переменные и функции, объявленные как `public`

3. Что называется конструктором?

- метод, имя которого совпадает с именем класса и который вызывается автоматически при создании объекта класса; +
- метод, имя которого совпадает с именем класса и который вызывается автоматически при объявлении класса (до создания объекта класса)
- метод, имя которого необязательно совпадает с именем класса и который вызывается при создании объекта класса
- метод, имя которого совпадает с именем класса и который необходимо явно вызывать из головной программы при объявлении объекта класса

4. Объект - это

- переменная, содержащая указатель на класс
- экземпляр класса; +
- класс, который содержит в себе данные и методы их обработки

5. Отметьте правильные утверждения

- конструкторы класса не наследуются; +
- конструкторов класса может быть несколько, их синтаксис определяется программистом;
- конструкторов класса может быть несколько, но их синтаксис должен подчиняться правилам перегрузки функций; +
- конструктор возвращает указатель на объект;
- конструктор не возвращает значение. +

6. Что называется деструктором?

- метод, который уничтожает объект
- метод, который удаляет объект
- **метод, который освобождает память, занимаемую объектом; +**
- системная функция, которая освобождает память, занимаемую объектом

7. Выберите правильные утверждения

- **у конструктора могут быть параметры; +**
- конструктор наследуется, но должен быть перегружен;
- конструктор должен явно вызываться всегда перед объявлением объекта;
- **конструктор вызывается автоматически при объявлении объекта; +**
- объявление каждого класса должно содержать свой конструктор;
- **если конструктор не создан, компилятор создаст его автоматически. +**

8. Выберите правильные утверждения

- деструктор - это метод класса, применяемый для удаления объекта;
- *** деструктор - это метод класса, применяемый для освобождения памяти, занимаемой объектом; +**
- деструктор - это отдельная функция головной программы, применяемая для освобождения памяти, занимаемой объектом;
- *** деструктор не наследуется; +**
- деструктор наследуется, но должен быть перегружен.

9. Что называется наследованием?

- *** это механизм, посредством которого производный класс получает элементы родительского и может дополнять либо изменять их свойства и методы; +**
- это механизм переопределения методов базового класса;
- это механизм, посредством которого производный класс получает все поля базового класса;
- это механизм, посредством которого производный класс получает элементы родительского, может их дополнить, но не может переопределить.

10. Выберите правильное объявление производного класса

- class MoreDetails:: Details;
- class MoreDetails: public class Details;
- *** class MoreDetails: public Details; +**
- class MoreDetails: class(Details);

11. Выберите правильные утверждения:

- если элементы класса объявлены как private, то они доступны только наследникам класса, но не внешним функциям;
- *** если элементы класса объявлены как private, то они недоступны ни наследникам класса, ни внешним функциям;**
- если элементы объявлены как public, то они доступны наследникам класса, но не внешним функциям;
- *** если элементы объявлены как public, то они доступны и наследникам класса, и внешним функциям.**

12. Возможность и способ обращения производного класса к элементам базового определяется

- ключами доступа: private, public, protected в теле производного класса;
- только ключом доступа protected в заголовке объявления производного класса;

- * **ключами доступа: private, public, protected в заголовке объявления производного класса; +**
- ключами доступа: private, public, protected в теле базового класса.

13. Выберите правильные соответствия между спецификатором базового класса, ключом доступа в объявлении производного класса и правами доступа производного класса к элементам базового

- ключ доступа - public; в базовом классе: private; права доступа в производном классе – protected;
- * **ключ доступа - любой; в базовом классе: private; права доступа в производном классе - нет прав; +**
- * **ключ доступа - protected или public ; в базовом классе: protected; права доступа в производном классе – protected; +**
- ключ доступа - private; в базовом классе: public; права доступа в производном классе – public;
- * **ключ доступа – любой; в базовом классе: public; права доступа в производном классе – такие же, как ключ доступа. +**

14. Дружественная функция - это

- функция другого класса, среди аргументов которой есть элементы данного класса;
- * **функция, объявленная в классе с атрибутом friend, но не являющаяся членом класса; +**
- функция, являющаяся членом класса и объявленная с атрибутом friend;
- функция, которая в другом классе объявлена как дружественная данному.

15. Выберите правильные утверждения:

- * **одна функция может быть дружественной нескольким классам;**
- дружественная функция не может быть обычной функцией, а только методом другого класса;
- * **дружественная функция объявляется внутри класса, к элементам которого ей нужен доступ;**
- дружественная функция не может быть методом другого класса.

16. Шаблон функции - это...

- * **определение функции, в которой типу обрабатываемых данных присвоено условное обозначение; +**
- прототип функции, в котором вместо имен параметров указан условный тип;
- определение функции, в котором указаны возможные варианты типов обрабатываемых параметров;
- определение функции, в котором в прототипе указан условный тип, а в определении указаны варианты типов обрабатываемых параметров.

17. Выберите правильные утверждения:

- * **по умолчанию члены класса имеют атрибут private; +**
- по умолчанию члены класса имеют атрибут public;
- члены класса имеют доступ только к элементам public;
- * **элементы класса с атрибутом private доступны только членам класса. +**

18. Переопределение операций имеет вид:

- имя_класса, ключевое слово operation, символ операции;
- * **имя_класса, ключевое слово operator, символ операции, в круглых скобках могут быть указаны аргументы; +**

- имя_класса, ключевое слово operator, список аргументов;
- имя_класса, два двоеточия, ключевое слово operator, символ операции.

19. Для доступа к элементам объекта используются:

- * **при обращении через имя объекта – точка, при обращении через указатель – операция «->»;** +
 - при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «точка»;
 - при обращении через имя объекта – точка, при обращении через указатель – два двоеточия;
 - при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «->».

20. Полиморфизм – это:

- * **средство, позволяющее использовать одно имя для обозначения действий, общих для родственных классов;** +
 - средство, позволяющее в одном классе использовать методы с одинаковыми именами;
 - средство, позволяющее в одном классе использовать методы с разными именами для выполнения одинаковых действий;
 - средство, позволяющее перегружать функции для работы с разными типами или разным количеством аргументов.

21. Полиморфизм реализован через механизмы:

- * **перегрузки функций, виртуальных функций, шаблонов;** +
 - перегрузки функций, наследования методов, шаблонов;
 - наследования методов, виртуальных функций, шаблонов;
 - перегрузки функций, наследования, виртуальных функций.

22. Виртуальными называются функции:

- * **функции базового класса, которые могут быть переопределены в производном классе;** +
 - функции базового класса, которые не используются в производном классе;
 - функции базового класса, которые не могут быть переопределены в базовом классе;
 - функции производного класса, переопределенные относительно базового класса.

23. Выберите правильный вариант выделения динамической памяти под переменную X типа float:

- * **float *ptr = new float; X = *ptr;** +
 - float & ptr = new float; X = & ptr;
 - float * ptr = &X; X = new float;

24. Полиморфизм в объектно-ориентированном программировании реализуется:

- * **через механизмы перегрузки (функций и операций), виртуальные функции и шаблоны;** +
 - через механизмы перегрузки (функций и операций) и шаблоны;
 - через виртуальные функции и шаблоны;
 - через механизмы перегрузки (функций и операций) и виртуальные функции.

25. Дано определение класса. Описать название класса, переменные и методы.

```
class monstr {
int health, armo;
monstr (int he, int arm);
```

```
public:  
monstr(int he=50, int arm=10);  
int color;  
}
```

Общая сумма баллов – 10 баллов.
За каждый правильный ответ – 0,4 балла.

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)**

Кафедра Транспорта и технологий нефтегазового комплекса

Комплект тем сообщений (рефератов)

1. Сортировка методом пузырька (обменом).
2. Сортировка выбором. Сравнительный анализ методов сортировки.
3. Сортировка вставками. Сравнительный анализ методов сортировки.
4. Метод быстрой сортировки. Сравнение итерации и рекурсии.
5. Последовательный и бинарный поиск в массиве.
6. Рекурсивный алгоритм бинарного поиска. Сравнение итерации и рекурсии.
7. Алгоритм Евклида (нахождение НОД): итерационный и рекурсивный. Алгоритм нахождения НОК. Сравнение итерации и рекурсии.
8. Сортировка подсчетом. Множество как тип данных.
9. Генерирование перестановок.
10. Переборные алгоритмы. Задача о 8-ми ферзях.
11. Алгоритм приближенного решения уравнения методом бинарного деления.
12. Рекурсия в задаче о ханойских башнях.
13. Алгоритмы моделирования многочленов на базе массивов.
14. Матрица как двумерный массив: процедуры ввода, вывода и заполнения матрицы.
15. Алгоритмы основных действий с матрицами.
16. Алгоритм построения магических квадратов.
17. Алгоритмы со строками: проверка правильности расстановки скобок в строке.
18. Алгоритм приближенного вычисления бесконечной суммы (см. цикл).
19. Алгоритм поиска простых чисел (Решето Эратосфена).
20. Алгоритмы основных процедур и функций (подпрограммы) для работы с массивами.
21. Алгоритмы длинной арифметики: сложение длинных чисел.
22. Рекурсивный алгоритм вычисления натуральной степени числа.
23. Поиск кратчайшего пути в неориентированном графе.
24. Построение остовного дерева неориентированного графа.
25. Сравнительный анализ методов сортировки.

Требования к содержанию и оформлению

Объем сообщения (реферата) – 10-12 страниц текста, оформленного в соответствии с указанными ниже требованиями:

- титульный лист;
- содержание;
- введение;
- основная часть;
- заключение;
- список использованных источников;
- приложения (при необходимости).

Оформляется задание письменно, оно может включать элементы наглядности (иллюстрации, демонстрацию).

Регламент времени на озвучивание сообщения – до 15 мин.

Этапы работы над сообщением (рефератом):

1. Подбор и изучение основных источников по теме, указанных в данных рекомендациях.
2. Составление списка использованных источников.
3. Обработка и систематизация информации.
4. Написание сообщения (реферата).
5. Публичное выступление и защита сообщения (реферата).

Критерии оценивания сообщения (реферата):

- 1) актуальность темы;
- 2) соответствие содержания теме;
- 3) глубина проработки материала;
- 4) грамотность и полнота использования источников;
- 5) наличие элементов наглядности;
- 6) устный рассказ.

Критерии оценки:

8–9 баллов – все критерии выполнены на 90-100% (или выполнены только 5 критериев);

5–7 баллов – все критерии выполнены на 60-89% (или выполнены только 4 критерия);

0–4 баллов – все критерии выполнены на 0-59% (или выполнены 3 и менее критериев).

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
НОЯБРЬСКИЙ ИНСТИТУТ НЕФТИ И ГАЗА
(ФИЛИАЛ) ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ТЮМЕНСКИЙ ИНДУСТРИАЛЬНЫЙ УНИВЕРСИТЕТ»
(Филиал ТИУ в г.Ноябрьске)**

Кафедра Транспорта и технологий нефтегазового комплекса

Перечень вопросов к экзамену

1. Постановка задачи. Хорошо и плохо поставленные задачи.
2. Модель задачи. Адекватность модели. Тестирование.
3. Состав языка.
4. Основные типы данных, объявление переменных.
5. Операции и выражения.
6. Базовые конструкции: оператор «выражение», операторы ветвления, операторы цикла, операторы передачи управления.
7. Объявление и описание функций.
8. Глобальные и локальные переменные.
9. Передача параметров в функции по значению и по ссылке.
10. Программирование рекурсивных алгоритмов.
11. Концепция модульного программирования.
12. Реализация рекурсии.
13. Концепция типа. Классификация типов данных.
14. Массивы: одномерные и многомерные.
15. Операции с массивами, передача массивов как параметров функций.
16. Алгоритмы сортировки и поиска в массивах.
17. Строка как массив символов.
18. Библиотечные функции обработки строк.
19. Применение файлов для ввода-вывода массивов.
20. Виды файлов.
21. Объявление файловой переменной.
22. Функции работы с файлами.
23. Структуры (struct/record): объявление, инициализация, доступ к полям.
24. Массивы структур. Иерархия структур.
25. Передача структур как параметров функций.
26. Описание класса.
27. Объявления переменных (объектов) типа данного класса.
28. Конструкторы и деструкторы класса.
29. Объектно-ориентированное программирование (ООП).
30. Реализация ООП в языках программирования.

Критерии оценки:

91-100 баллов (оценка «отлично») выставляется обучающемуся, глубоко и прочно усвоившему материал, исчерпывающе, грамотно и логически стройно его излагающего. Представлена схема (если в ответе на вопросе есть конструктивные элементы) Соответствующие знание, умения и владение сформированы полностью.

76-90 баллов (оценка «хорошо») выставляется обучающемуся, твердо знающему материал, грамотно и по существу излагающего его. Обучающийся не допускает существенных не-

точностей в ответе на вопросы. Соответствующие знание, умения и владение сформированы в целом полностью, но содержат отдельные пробелы.

61-75 баллов (оценка «удовлетворительно») выставляется обучающемуся, который имеет знания только основного материала, но не усвоил его детали, допускает неточности, недостаточно правильные формулировки, нарушения последовательности в изложении материала. Обучающийся показывает общее, но не структурированное знание, в целом успешное, но не систематическое умение и владение соответствующих компетенций.

0-60 баллов (оценка «не удовлетворительно») выставляется обучающемуся, который не усвоил значительной части материала, допускает существенные ошибки. Обучающийся показывает фрагментарные знания (или их отсутствие), частично освоенное умение (или его отсутствие), фрагментарное применение навыка (или его отсутствие) соответствующих компетенций.