

Документ подписан простой электронной подписью
Информация о владельце:
ФИС: Комарова Светлана Юриевна
Должность: Проректор по образовательной деятельности
Дата подписания: 05.09.2024 12:34:50
Уникальный программный ключ:
43ba42f5deae4116bbfcb9ac98e39108031227e81add207cbee4149f2098d7a~

**Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Омский государственный аграрный университет имени П.А. Столыпина»
Экономический факультет**

ОПОП по направлению подготовки
09.04.02 Информационные системы и технологии

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине**

Б1.О.11 Теоретические основы программирования

Направленность (профиль) «Информационные системы и технологии»

Обеспечивающая преподавание дисциплины кафедра	Кафедра экономики, бухгалтерского учета и финансового контроля
Разработчик, канд. экон. наук, доцент	Е.Е. Голова

ВВЕДЕНИЕ

1. Фонд оценочных средств по дисциплине является обязательным обособленным приложением к Рабочей программе.

2. Фонд оценочных средств является составной частью нормативно-методического обеспечения системы оценки качества освоения обучающимися указанной дисциплины.

3. При помощи ФОС осуществляется контроль и управление процессом формирования обучающимися компетенций, из числа предусмотренных ФГОС ВО в качестве результатов освоения дисциплины.

4. Фонд оценочных средств по дисциплине включает в себя: оценочные средства, применяемые для входного контроля; оценочные средства, применяемые в рамках индивидуализации выполнения и контроля фиксированных видов ВАРС; оценочные средства, применяемые для текущего контроля и оценочные средства, применяемые при промежуточной аттестации по итогам изучения дисциплины.

5. Разработчиками фонда оценочных средств по дисциплине являются преподаватели кафедры экономики, бухгалтерского учёта и финансового контроля обеспечивающей изучение обучающимися дисциплины в университете. Содержательной основой для разработки ФОС послужила Рабочая программа дисциплины.

1. ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ
учебной дисциплины модуля, персональный уровень достижения которых проверяется с
использованием представленных в п. 3 оценочных средств

Компетенции, в формировании которых задействована дисциплина		Код и наименование индикатора достижений компетенции	Компоненты компетенций, формируемые в рамках данной дисциплины (как ожидаемый результат ее освоения)		
код	наименование		знать и понимать	уметь делать (действовать)	владеть навыками (иметь навыки)
Общепрофессиональные компетенции					
ОПК-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач.	ИД-1 _{опк-2} Знает современные информационно-коммуникационные и интеллектуальные технологии, инструментальные среды, программно-технические платформы для решения профессиональных задач	Основные компьютерные технологии (языки, библиотеки, инструменты), используемые для решения прикладных задач; основные этапы решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.	Основными требованиями к разработке оптимальных алгоритмов, реализацию основных модулей программного продукта; Особенности компьютерного моделирования с использованием объектно-ориентированных технологий.	Основные алгоритмы, методы и принципы построения программных продуктов на языке программирования; особенности компьютерного моделирования с использованием объектно-ориентированных технологий.
		ИД-2 _{опк-2} Обосновывает выбор современных информационно-коммуникационных и интеллектуальных технологий в целях разработки оригинальных программных средств для решения профессиональных задач	Выбор и разработка оптимального алгоритма для его дальнейшей реализации при решении конкретных задач.	Разрабатывает алгоритмы для его дальнейшей реализации при решении конкретных задач на языке программирования, их отладка и тестирование с использованием объектно-ориентированных технологий.	Разработки прикладных программ на языке программирования, их отладка и тестирование с использованием объектно-ориентированных технологий.
		ИД-3 _{опк-2} Разрабатывает оригинальные программные средства для решения профессиональных задач, в том числе с использованием современных информационно-коммуникационных и интеллектуальных технологий	Основные конструкции языков программирования по разработке прикладных программ.	Реализует математические алгоритмы в виде законченных программных модулей моделирования процессов с использованием объектно-ориентированных технологий на языке программирования	Разрабатывает программы с использованием современных информационно-коммуникационных и интеллектуальных технологий, для решения профессиональных задач.

**ЧАСТЬ 2. ОБЩАЯ СХЕМА ОЦЕНИВАНИЯ ХОДА И РЕЗУЛЬТАТОВ ИЗУЧЕНИЯ
ДИСЦИПЛИНЫ**

Общие критерии оценки и реестр применяемых оценочных средств

**2.1. Общие критерии оценки хода и результатов
изучения учебной дисциплины**

1. Формальный критерий получения обучающимися положительной оценки по итогам изучения дисциплины:	
1.1 Предусмотренная программа изучения дисциплины обучающимся выполнена полностью до начала процесса промежуточной аттестации	1.2 По каждой из предусмотренных программой видов работ по дисциплине обучающийся успешно отчитался перед преподавателем, демонстрируя при этом должный (не ниже минимально приемлемого) уровень сформированности элементов компетенций
2. Группы неформальных критериев качественной оценки работы обучающегося в рамках изучения дисциплины:	
2.1 Критерии оценки качества хода процесса изучения обучающимся программы дисциплины (текущей успеваемости)	2.2. Критерии оценки качества выполнения конкретных видов ВАРС
2.3 Критерии оценки качественного уровня итоговых результатов изучения дисциплины	2.4. Критерии аттестационной оценки качественного уровня результатов изучения дисциплины

2.2 РЕЕСТР
элементов фонда оценочных средств по учебной дисциплине

Группа оценочных средств	Оценочное средство или его элемент
	Наименование
1. Средства для текущего контроля	Тестовые вопросы для проведения текущего контроля
	Критерии оценки ответов на тестовые вопросы входного контроля
2. Средства для индивидуализации выполнения, контроля фиксированных видов ВАРС	Перечень тем для выполнения презентаций.
	Критерии оценки индивидуальных результатов выполнения презентации
	Перечень заданий для выполнения РАР1.
	Критерии оценки индивидуальных результатов выполнения РАР 1
	Перечень заданий для выполнения РАР 2.
	Критерии оценки индивидуальных результатов выполнения РАР 2
3. Средства для текущего контроля	Вопросы для самостоятельного изучения темы
	Общий алгоритм самостоятельного изучения темы
	Критерии оценки самостоятельного изучения темы
	Критерии оценки самоподготовки по темам лабораторных занятий
4. Средства для промежуточной аттестации по итогам изучения дисциплины	Тестовые вопросы для проведения итогового контроля (экзамена)
	Экзаменационная программа по учебной дисциплине
	Плановая процедура проведения экзамена
	Критерии оценки ответов на тестовые вопросы итогового контроля

2.4 Описание показателей, критериев и шкал оценивания и этапов формирования компетенций в рамках дисциплины

Индекс и название компетенции	Код индикатора достижений компетенции	Индикаторы компетенции	Показатель оценивания – знания, умения, навыки (владения)	Уровни сформированности компетенций				Формы и средства контроля формирования компетенций
				компетенция не сформирована	минимальный	средний	высокий	
				Оценки сформированности компетенций				
				2	3	4	5	
				Оценка «неудовлетворительно»	Оценка «удовлетворительно»	Оценка «хорошо»	Оценка «отлично»	
				Характеристика сформированности компетенции				
				Компетенция в полной мере не сформирована. Имеющихся знаний, умений и навыков недостаточно для решения практических (профессиональных) задач	Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач	Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных практических (профессиональных) задач	Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач	
Критерии оценивания								
ОПК-2	ИД-1 _{ОПК-2}	Полнота знаний	Основные компьютерные технологии (языки, библиотеки, инструменты), используемые для решения прикладных задач; основные этапы решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.	Не знает основные компьютерные технологии (языки, библиотеки, инструменты), используемые для решения прикладных задач; основные этапы решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.	Знает частично основные компьютерные технологии (языки, библиотеки, инструменты), используемые для решения прикладных задач; основные этапы решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.	Знает в недостаточной степени основные компьютерные технологии (языки, библиотеки, инструменты), используемые для решения прикладных задач; основные этапы решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.	Знает основные компьютерные технологии (языки, библиотеки, инструменты), используемые для решения прикладных задач; основные этапы решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.	Тестирование, РГР 1, РГР 2, создание презентации

		Наличие умений	Основными требованиями к разработке оптимальных алгоритмов, реализацию основных модулей программного продукта; Особенности компьютерного моделирования с использованием объектно-ориентированных технологий.	Допускает грубые ошибки при разработке оптимальных алгоритмов и реализации основных модулей программного продукта;	Допускает ошибки при разработке оптимальных алгоритмов.	Допускает ошибки в разработке основных оптимальных алгоритмов и реализации основных модулей программного продукта;	Умеет разрабатывать оптимальные алгоритмы и реализует основные модули программного продукта;	Тестирование, РГР 1, РГР 2, создание презентации
		Наличие навыков (владение опытом)	Основные алгоритмы, методы и принципы построения программных продуктов на языке программирования; особенности компьютерного моделирования с использованием объектно-ориентированных технологий.	Не владеет основными алгоритмами, методами и принципами построения программных продуктов на языке программирования; особенностями компьютерного моделирования с использованием объектно-ориентированных технологий.	Частично владеет основными алгоритмами, методами и принципами построения программных продуктов на языке программирования; особенностями компьютерного моделирования с использованием объектно-ориентированных технологий.	Допускает ошибки при разработке алгоритмов, методов и принципов построения программных продуктов на языке программирования; особенности компьютерного моделирования с использованием объектно-ориентированных технологий.	Владеет разработкой алгоритмов, методов и принципов построения программных продуктов на языке программирования; особенностями компьютерного моделирования с использованием объектно-ориентированных технологий.	
	ИД-2 _{ОПК-2}	Полнота знаний	Выбор и разработка оптимального алгоритма для его дальнейшей реализации при решении конкретных задач.	Не знает, как реализовать алгоритмы для конкретных задач при их решении.	Знает частично, как выбрать и разработать оптимальный алгоритм для реализации при решении конкретных задач.	Знает в недостаточной степени как выбрать и разработать оптимальный алгоритм для его дальнейшей реализации при решении конкретных задач.	Знает, как выбрать и разработать оптимальный алгоритм для его дальнейшей реализации при решении конкретных задач.	Тестирование, опрос, РАР 1, РАР 2, создание презентации

		Наличие умений	Разрабатывает алгоритмы для его дальнейшей реализации при решении конкретных задач на языке программирования, их отладка и тестирование с использованием объектно-ориентированных технологий.	Допускает грубые ошибки при разработке алгоритмов для его реализации при решении конкретных задач на языке программирования, отладке и тестированию с использованием объектно-ориентированных технологий.	Допускает ошибки при разработке алгоритмов для их дальнейшей реализации при решении конкретных задач.	Допускает ошибки при разработке при решении конкретных задач на языке программирования, отладке и тестировании с использованием объектно-ориентированных технологий.	Умеет разрабатывать алгоритмы для их дальнейшей реализации при решении конкретных задач на языке программирования. Умеет выполнять отладку и тестирование с использованием объектно-ориентированных технологий.	Тестирование, опрос, РАР 1, РАР 2, создание презентации
		Наличие навыков (владение опытом)	Разработки прикладных программ на языке программирования, их отладка и тестирование с использованием объектно-ориентированных технологий.	Не владеет основными навыками разработки прикладных программ на языке программирования, их отладки и тестировании с использованием объектно-ориентированных технологий.	Частично владеет основными навыками разработки прикладных программ на языке программирования.	Владеет основными навыками разработки прикладных программ на языке программирования, их отладкой и тестированием с использованием объектно-ориентированных технологий.	Владеет навыками разработки прикладных программ на языке программирования, их отладка и тестирование с использованием объектно-ориентированных технологий.	
	ИД-3 _{ОПК-2}	Полнота знаний	Основные конструкции языков программирования по разработке прикладных программ.	Не знает основные конструкции языков программирования по разработке прикладных программ.	Знает частично основные конструкции языков программирования по разработке прикладных программ.	Знает в недостаточной степени основные конструкции языков программирования по разработке прикладных программ.	Знает основные конструкции языков программирования по разработке прикладных программ.	Тестирование, опрос,
		Наличие умений	Реализует математические алгоритмы в виде законченных программных модулей моделирования процессов с использованием объектно-ориентированных технологий на языке программирования	Допускает грубые ошибки при реализации математических алгоритмов в виде законченных программных модулей моделирования процессов с использованием объектно-ориентированных технологий на языке программирования	Допускает ошибки при реализации математических алгоритмов в виде законченных программных модулей моделирования процессов с использованием объектно-ориентированных технологий на языке программирования	Допускает при реализации математических алгоритмов в виде законченных программных модулей моделирования процессов с использованием объектно-ориентированных технологий на языке программирования	Умеет реализовать математические алгоритмы в виде законченных программных модулей моделирования процессов с использованием объектно-ориентированных технологий на языке программирования	

			языке программирован ия					РАР 1, РАР 2, создание презентации
		Наличие навыков (владение опытом)	Разрабатывает программы с использованием современных информационно- коммуникационн ых и интеллектуальн ых технологий, для решения профессиональн ых задач.	Не владеет основными навыками разработки программ с использованием современных информационно- коммуникационных и интеллектуальных технологий, для решения профессиональных задач.	Частично владеет разработкой программ с использованием современных информационно- коммуникационных и интеллектуальных технологий, для решения профессиональных задач.	Владеет основными навыками разработки программ с использованием современных информационно- коммуникационных и интеллектуальных технологий, для решения профессиональных задач.	Владеет навыками разработки программ с использованием современных информационно- коммуникационных и интеллектуальных технологий, для решения профессиональных задач.	Тестирование, опрос, РАР 1, РАР 2, создание презентации

ЧАСТЬ 3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков, характеризующих этапы формирования компетенций

Часть 3.1. Типовые контрольные задания, необходимые для оценки знаний, умений, навыков

**3.1.1 . Средства
для индивидуализации выполнения, контроля фиксированных видов ВАРС**

3.1.1.1 Выполнение и сдача электронной презентации

3.1.1.2 Место электронной презентации в структуре дисциплины

Разделы дисциплины, освоение которых обучающимися сопровождается или завершается выполнением электронной презентации		Компетенции, формирование/развитие которых обеспечивается в ходе выполнения электронной презентации
№	Наименование	
1-16	Технологии программирования. Назначение и основные особенности систем программирования.	ОПК - 2

3.1.1.3 Перечень примерных тем электронной презентации

ПРИМЕРНАЯ ТЕМАТИКА

1. Основные конструкции языка C++. Задачи и особенности прикладного программирования. Структура программы на языке C++. Размещение программы и данных в памяти. Переменные; значение, указатель, ссылка. Динамическое размещение данных в памяти.
2. Реализация вычислительных операций. Арифметические и логические выражения. Основные языковые конструкции (условные, циклические, селективные конструкции). Функции объявление и определение. Передача аргументов в функции. Стандартная библиотека функция языка C++. Библиотека стандартного ввода/вывода. Форматированный ввод/вывод. Файловые потоки.
3. Составные типы данных (массивы, контейнеры). Массивы – как пример гомогенной структуры данных, размещение в памяти, доступ к элементам.
4. Составные типы данных (массивы, контейнеры). Одномерные и многомерные массивы.
5. Составные типы данных (массивы, контейнеры). Контейнеры библиотеки STL.
6. Абстрактные типы данных. Классы. Инкапсуляция. Скрытие данных и видимость членов класса. Конструктор и деструктор.
7. Наследование. Виртуальные функции и абстрактные базовые классы. Множественное наследование.
8. Полиморфизм. Перегрузка функций. Перегрузка операторов (унарного, бинарного, особые случаи). Параметрический полиморфизм. Шаблоны функций. Шаблоны классов.
9. Контейнеры и итераторы в библиотеке STL (Standard Template Library). Вектор очереди. Стек. Список. Ассоциативные массивы. Алгоритмы. Объекты-функции и предикаты.
10. Вычисление обратной матрицы.
11. Генерация псевдослучайных чисел и анализ распределений случайных величин.
12. Графический метод решения задачи линейного программирования.
13. Группировка (кластеризация) экономических объектов.
14. Диаграмма Ганта и ее экономические приложения.
15. Задача вычисления обратной матрицы и ее экономико-математические приложения.
16. Интерполирование таблично заданной функции.
17. Кодирование и декодирование произвольных сообщений.
18. Компьютерная мультипликация (спрайты).
19. Краткосрочное прогнозирование экономических показателей методом экспоненциального сглаживания.
20. Кривые второго порядка. Метод наименьших квадратов и его экономико-математические приложения.
21. Минимизация функции методом градиентного спуска.

22. Минимизация функции методом случайного поиска.
23. Моделирование производственных функций.
24. Моделирование систем управления запасами.
25. Моделирование экономических показателей методом Монте-Карло.
26. Нахождение допустимого базисного решения задачи линейного программирования.
27. Нормальный закон распределения случайной величины.
28. Обработка и графическое представление данных о динамике курса рубля (доллара, евро).
29. Обработка статистической информации о посетителях Web-сайта.
30. Обработка экономической информации и среднесрочное прогнозирование.
31. Определение равновесного выпуска (методами простой итерации, Гаусса – Зейделя, пошагового агрегирования).
32. Определение равновесных цен.
33. Оценка доходности облигаций.
34. Оценка производственных функций (линейным и нелинейным методом наименьших квадратов).
35. Оценка стоимости акций.
36. Оценка стоимости опционов.
37. Построение дерева решений.
38. Построение интерполяционного многочлена Лагранжа.
39. Построение кривых безразличия (планов потребления).
40. Построение кривых второго порядка (в прямоугольной системе координат).
41. Построение кубических сплайнов.
42. Преобразование системы векторов методом Гаусса.
43. Приближенное вычисление определенных интегралов методом прямоугольников, трапеций, парабол (Симпсона).
44. Проверка статистических гипотез.
45. Программа — графический редактор.
46. Разработка многоязычного пользовательского интерфейса.
47. Расчет и анализ показателей оценки доходности ценных бумаг.
48. Расчет и анализ рисков инвестиционного проекта.
49. Расчет и анализ финансовых показателей предприятия.
50. Расчет показателей эффективности инвестиций с учетом риска.
51. Решение алгебраических уравнений комбинированным методом.
52. Решение системы линейных уравнений методом Крамера.
53. Рыночное регулирование спроса и предложения в условиях олигополистической конкуренции (случаи дуополии Курно, Штакельберга, Бертрана).
54. Рыночное регулирование спроса и предложения в условиях совершенной конкуренции (паутинообразная модель, модель общего равновесия, двухсекторная модель).
55. Сетевые графики и их экономические приложения.
56. Транспортная задача и ее экономические приложения.
57. Формирование оптимального портфеля ценных бумаг.
58. Численное дифференцирование таблично заданной функции.
59. Численное решение алгебраических уравнений (методом половинного деления, касательных).
60. Численное решение двойственной задачи линейного программирования.
61. Численное решение задачи линейного программирования.
62. Численное решение обыкновенных дифференциальных уравнений (методом Эйлера, Рунге – Кутты).....

Процедура выбора темы обучающимся

Учебные цели, на достижение которых ориентировано выполнение презентации: получить целостное представление об основных компьютерных технологиях (языки, библиотеки, инструменты), используемых для решения прикладных задач; основных этапах решения задач на компьютере, порядок разработки, отладки, тестирования и документирования программного продукта.

Учебные задачи, которые должны быть решены обучающимся в рамках выполнения презентации:

- детальное рассмотрение наиболее актуальных вопросов программирования;
- формирование и отработка навыков составления программ, накопление опыта работы с научной литературой, подбора и анализа фактического материала;

- совершенствование в изложении своих мыслей, критики, самостоятельного построения структуры работы, постановки задач, раскрытие основных вопросов, умение сформулировать логические выводы и предложения.

Этапы работы над презентацией

Выбор темы. Очень важно правильно выбрать тему. Выбор темы не должен носить формальный характер, а иметь практическое и теоретическое обоснование.

Автор должен осознанно выбрать тему с учетом его познавательных интересов или он может увязать ее с темой будущей магистерской работы. В этом случае магистранту предоставляется право самостоятельного (с согласия преподавателя) выбора темы из списка тем, рекомендованных кафедрой по данной дисциплине (см. выше). При этом весьма полезными могут оказаться советы и обсуждение темы с преподавателем, который может оказать помощь в правильном выборе темы и постановке задач.

Если интересующая тема отсутствует в рекомендательном списке, то по согласованию с преподавателем обучающемуся предоставляется право самостоятельно предложить тему, раскрывающую содержание изучаемой дисциплины. Тема не должна быть слишком общей и глобальной, так как небольшой объем работы (до 20 слайдов) не позволит раскрыть ее.

При выборе темы необходимо учитывать полноту ее освещения в имеющейся научной литературе. Для этого можно воспользоваться тематическими каталогами библиотек и библиографическими указателями литературы, периодическими изданиями и ежемесячными указателями психолого-педагогической литературы, либо справочно-библиографическими ссылками изданий посвященных данной теме.

После выбора темы составляется список изданной по теме (проблеме) литературы, опубликованных статей, необходимых справочных источников.

Знакомство с любой научной проблематикой следует начинать с освоения имеющейся основной научной литературы. При этом следует сразу же составлять библиографические выходные данные (автор, название, место и год издания, издательство, страницы) используемых источников. Названия работ иностранных авторов приводятся только на языке оригинала.

Начинать знакомство с избранной темой лучше всего с чтения обобщающих работ по данной проблеме, постепенно переходя к узкоспециальной литературе.

На основе анализа прочитанного и просмотренного материала по данной теме следует составить тезисы по основным смысловым блокам, с пометками, собственными суждениями и оценками. Предварительно подобранный в литературных источниках материал может превышать необходимый объем презентации, но его можно использовать для составления плана.

Составление плана. Автор по предварительному согласованию с преподавателем может самостоятельно составить план, с учетом замысла работы, либо взять за основу рекомендуемый план, приведенный в данных методических указаниях по соответствующей теме. Правильно построенный план помогает систематизировать материал и обеспечить последовательность его изложения.

Наиболее традиционной является следующая структура:

Титульный лист.

Оглавление (план, содержание).

Введение.

Глава 1 (полное наименование главы).

1.1. (полное название параграфа, пункта);

1.2. (полное название параграфа, пункта).

Глава 2 (полное наименование главы).

2.1. (полное название параграфа, пункта);

2.2. (полное название параграфа, пункта).

Заключение (или выводы).

Список использованной литературы.

Приложения (по усмотрению автора).

Основная часть

Первый слайд заполняется по единой форме (Приложение 1).

Оглавление (план, содержание) включает названия всех разделов (пунктов плана) презентации и номера страниц, указывающие начало этих разделов в тексте презентации.

Введение. В этой части обосновывается актуальность выбранной темы, формулируются цели работы и основные вопросы, которые предполагается раскрыть в презентации, указываются используемые материалы и дается их краткая характеристика с точки зрения полноты освещения избранной темы. Объем введения не должен превышать 1-1,5 слайда.

Основная часть презентации может быть представлена одной или несколькими главами, которые могут включать 2-3 слайда (подпункта, раздела).

Здесь достаточно полно и логично излагаются главные положения в используемых источниках, раскрываются все пункты плана с сохранением связи между ними и последовательности перехода от одного к другому.

Автор должен следить за тем, чтобы изложение материала точно соответствовало цели и названию главы (параграфа). Материал в презентации рекомендуется излагать своими словами, не допуская дословного переписывания из литературных источников. В тексте обязательны ссылки на первоисточники, т.е. на тех авторов, у которых взят данный материал в виде мысли, идеи, вывода, числовых данных, таблиц, графиков, иллюстраций и пр.

Работа должна быть написана грамотным литературным языком. Сокращение слов в тексте не допускается, кроме общеизвестных сокращений и аббревиатуры. Каждый раздел рекомендуется заканчивать кратким выводом.

Заключение (выводы). В этой части обобщается изложенный в основной части материал, формулируются общие выводы, указывается, что нового лично для себя вынес автор из работы над презентацией. Выводы делаются с учетом опубликованных в литературе различных точек зрения по проблеме рассматриваемой в теме презентации, сопоставления их и личного мнения автора презентации. Заключение по объему не должно превышать 1,5-2 слайда.

Приложения могут включать графики, таблицы, расчеты. Они должны иметь внутреннюю (собственную) нумерацию страниц.

Библиография (список литературы) здесь указывается реально использованная литература, периодические издания и электронные источники информации. Список составляется согласно правилам библиографического описания.

ШКАЛА И КРИТЕРИИ ОЦЕНКИ

Критерии оценивания презентаций складываются из требований к их созданию.

Название критерия	Оцениваемые параметры
Тема презентации	Соответствие темы программе учебного предмета, раздела
Дидактические и методические цели и задачи презентации	• Соответствие целей поставленной теме • Достижение поставленных целей и задач
Выделение основных идей презентации	• Соответствие целям и задачам • Содержание умозаключений • Вызывают ли интерес у аудитории • Количество (рекомендуется для запоминания аудиторией не более 4-5)
Содержание	• Достоверная информация об исторических справках и текущих событиях • Все заключения подтверждены достоверными источниками • Язык изложения материала понятен аудитории • Актуальность, точность и полезность содержания
Подбор информации для создания проекта – презентации	• Графические иллюстрации для презентации • Статистика • Диаграммы и графики • Экспертные оценки • Ресурсы Интернет • Примеры • Сравнения • Цитаты и т.д.
Подача материала проекта – презентации	• Хронология • Приоритет • Тематическая последовательность • Структура по принципу «проблема-решение»
Логика и переходы во время проекта – презентации	• От вступления к основной части • От одной основной идеи (части) к другой • От одного слайда к другому • Гиперссылки

Заключение	• Яркое высказывание - переход к заключению • Повторение основных целей и задач выступления • Выводы • Подведение итогов • Короткое и запоминающееся высказывание в конце
Дизайн презентации	• Шрифт (читаемость) • Корректно ли выбран цвет (фона, шрифта, заголовков) • Элементы анимации
Техническая часть	• Грамматика • Подходящий словарь • Наличие ошибок правописания и опечаток

КРИТЕРИИ И ШКАЛЫ ОЦЕНКИ, используемые при проверке и приёме электронной презентации	
Оцениваемая компонента электронной презентации и/или образовательных результатов работы над ней	Оценка по данной компоненте
а) Соответствие содержания презентации ее теме	Соответствует полностью/ не соответствует
б) Полнота и глубина раскрытия темы презентации	Высокая/достаточная/ приемлемая/ не приемлемая
в) Логика и глубина сделанных выводов	Высокая/достаточная/ приемлемая/ не приемлемая
г) Степень самостоятельности студента при подготовке презентации	Не вызывает сомнения/ Вызывает сомнения
д) Степень соблюдения студентом общих требований	Общие требования соблюдены полностью/ соблюдены на приемлемом уровне/ не соблюдены
– к оформлению презентации	
е) Уровень понимания студентом отражённого в презентации материала, проявленный на семинаре-конференции	Соответствует требуемому полностью / Находится на приемлемом уровне /
ж) Уровень коммуникативных навыков, продемонстрированный студентом на семинаре-конференции	Не соответствует минимально требуемому

Отлично	Оценка «отлично» по презентации присваивается за глубокое раскрытие темы, качественное оформление работы, содержательность презентации, за понимание студентом отражённого в презентации материала
Хорошо	Оценка «хорошо» по презентации присваивается при соответствии выше перечисленным критериям, но при наличии в содержании работы и ее оформлении небольших недочетов или недостатков в представлении результатов к защите
Удовлетворительно	Оценка «удовлетворительно» по презентации присваивается за неполное раскрытие темы, выводов и предложений, носящих общий характер, плохо подготовленное наглядное представление работы и затруднения при ответах на вопросы
Неудовлетворительно	Оценка «неудовлетворительно» по презентации присваивается за слабое и неполное раскрытие темы, несамостоятельность изложения материала, выводы и предложения, носящие общий характер, отсутствие наглядного представления работы и ответов на вопросы

3.1.2 Выполнение и сдача РАР 1

3.1.2.1 Место РАР 1 в структуре дисциплины

Разделы дисциплины, освоение которых обучающимися сопровождается или завершается выполнением РАР 1		Компетенции, формирование/развитие которых обеспечивается в ходе выполнения РАР 1
№	Наименование	
13	Основы объектно-ориентированного программирования. Проектирование форм в среде программирования Delphi	ОПК - 2

3.1.2.2 Перечень примерных тем РАР 1

ЗАДАНИЕ

Разработать программу для решения задачи, указанной в вашем варианте задания.

Каждая используемая при решении задачи структура данных должна быть оформлена как самостоятельный тип — класс с набором операций, необходимых для работы с этой структурой. Класс (или иерархия родственных классов) размещается в отдельном модуле.

В методах класса и вспомогательных подпрограммах необходимо предусмотреть обработку исключительных ситуаций, возникающих при работе со структурами данных.

Необходимо создать два варианта программы — отладочную и рабочую версии.

Рабочая версия предназначена для решения задачи задания.

Отладочный вариант демонстрирует работоспособность всех методов класса, вспомогательных процедур и функций работы с объектом при различных наборах входных данных. Наборы отладочных и рабочих исходных данных считываются из соответствующих текстовых файлов. Результаты выполнения отладочной и рабочей программ также помещаются в предназначенные для этого текстовые файлы.

ВАРИАНТЫ ЗАДАНИЙ РАР 1

1. Программа преобразования выражения из инфиксной формы в постфиксную

Программа выполняет следующее:

- считывает из внешнего файла последовательность выражений в инфиксной форме;
- преобразует форму представления выражения из полноскобочной инфиксной в постфиксную;
- осуществляет в процессе перевода синтаксический контроль правильности записи выражения;
- помещает результаты работы во внешний файл.

Исходное выражение содержит идентификаторы с числом буквенно-цифровых символов ≤ 6 , целые десятичные константы (≤ 32767) и операции $+$, $-$, $*$, $/$, $^{\wedge}$; операнды и операции отделяются друг от друга любым количеством пробелов.

При синтаксическом контроле проверяются: длина идентификатора (≤ 6), правильность его записи (первый символ — буква), значение константы (≤ 32767), наличие более одного подряд записанного знака операции, парность и правильность вложения скобок. Замечания:

- программа использует стек на динамических переменных;
- в программе предусмотреть булевские функции:

ТИеМат(81г:8Шд):Boo1ean; //строка идентификатор? ТИеОес(8^:8^пд):Boo1ean; // строка десятичная константа?

ТИеОрб(8^:81ппд):Boo1ean; //строка операнд?

ТИеОрс(81г:81ппд):Boo1ean; //строка операция?

2. Программа, имитирующая работу условной вычислительной машины

Программа выполняет следующее:

- считывает из внешнего файла последовательность выражений в постфиксной форме;
- генерирует последовательность команд условной вычислительной машины, необходимых для вычисления выражения, вычисляет с их помощью значение выражения и оставляет результат в регистре;
- переводит выражение из постфиксной записи в полноскобочную инфиксную;
- помещает результаты работы во внешний файл.

Исходное выражение содержит идентификаторы с числом буквенно-цифровых символов не больше шести, целые десятичные константы в диапазоне —32768 ... 32767 и операции +, —, *, /; операнды и операции отделяются друг от друга любым количеством пробелов.

Условная вычислительная машина имеет один регистр и следующую систему команд:

БЭ А — помещает операнд А в регистр;

ЭТ А — помещает содержимое регистра в переменную А;

АО А — прибавляет содержимое переменной А к регистру;

БВА — вычитает содержимое переменной А из регистра;

МБ А — умножает содержимое регистра на переменную А;

ЭУ А — делит содержимое регистра на переменную А.

Замечания:

- программа использует стек на динамических переменных;
- в качестве промежуточных переменных использовать имена Т1, Т2 и т.д.

3. Программа моделирования работы магазина самообслуживания

Составить программу моделирования работы магазина самообслуживания с заданным числом касс K . Каждый покупатель подходит в момент времени t , к свободной кассе или кассе с наиболее короткой очередью и требует для своего обслуживания времени t_2 (определяемого числом покупок).

Время прибытия покупателей и время обслуживания задаются в минутах задается с момента открытия магазина) и распределяются по нормальному закону с заданным средним значением M и отклонением 5 .

Программа должна определить среднее, минимальное и максимальное время обслуживания (время нахождения в очереди плюс время расчета) одного покупателя (для $K=3$ и $K=5$).

Исходные данные: K , M , P (число покупателей). Параметры t_2 для каждого покупателя определяются в программе с помощью генератора случайных чисел.

В отладочном варианте t_2 для каждого покупателя считывать из внешнего файла, сформированного вручную.

4. Программа выполнения арифметических операций с длинными числами

Составить программу алгебраического сложения и умножения длинных целых чисел с использованием двунаправленных связанных списков. Каждый элемент списка должен быть целым значением с максимально допустимым в используемой системе программирования количеством цифр.

Наборы исходных длинных целых чисел и операции, которые необходимо выполнять над ними, построчно вводятся из внешнего текстового файла в формате

число1_операция_число2 (_ — символ пробела) число3_операция_число4 ...

например:

- -1592678 + 278291 15 178591 *-1756 2997518 --324277

5. Программа пирамидальной сортировки

Составить программу упорядочения методом пирамидальной сортировки односвязного списка, каждый элемент которого состоит из двух полей: номер группы и фамилия студента.

Первоначально осуществить сортировку по возрастанию номера факультета (вторая цифра номера группы), затем по возрастанию номера курса (первая цифра), далее по возрастанию номера группы на факультете (третья цифра). Внутри каждой группы упорядочить фамилии по алфавиту.

Сравнить по быстродействию реализацию алгоритма пирамидальной сортировки для списка с полем значения — целым числом. С использованием такого списка построить демонстрационный вариант программы.

Размер рабочего исходного текстового файла — 100 записей.

6. Программа внутренней сортировки методом Неймана

Сравнить по быстродействию реализации метода Неймана (естественное слияние) для сортировки элементов односвязного списка и одномерного массива одинаковой длины.

В характеристики быстродействия включить число операций сравнения, число перестановок для массива и переключения связей для списка.

Типы данных и их значения выбрать произвольно. Длины массивов и списков принять равными 100, 200, 500 (для их формирования использовать генератор случайных чисел).

7. Программа быстрой сортировки

Оценить характеристики быстрой сортировки (число сравнений ключей и число перестановок элементов) для сортировки:

- 1) упорядоченного массива;
- 2) случайного массива;
- 3) упорядоченного в обратном порядке массива.

Размер сортируемого массива: 10, 50, 100, 500 (для формирования использовать генератор случайных чисел).

8. Программа поразрядной сортировки

Оценить характеристики метода поразрядной сортировки числового массива с использованием связанных списков. Сортировка выполняется сначала для младших значащих цифр, затем полученные списки сливаются в файл и сортируются по следующей значащей цифре, и т.д. до старшей цифры. В характеристики быстродействия включить: число операций сравнения, число переключений связей для элементов списков (аналог числа перестановок). Исходный массив состоит из 500 случайных чисел, равномерно распределенных в диапазоне от 0 до 9999.

Сравнить по быстродействию поразрядную сортировку с разбиением файла на 10 связанных списков по старшей значащей цифре с включением элементов в список в порядке, не нарушающем упорядоченность списка.

9. Программа индексно-последовательного поиска

Составить программу формирования таблицы символических имен в виде индексно-последовательного файла и последующего поиска имени в таблице.

Сравнить по среднему времени поиска (аналог — среднее число просмотренных при поиске элементов) реализации индексно-последовательного поиска:

- 1) число индексов: 1 и 2;
- 2) размер индекса: 3 и 10.

Файл с исходным списком символических имен сформировать с помощью подпрограммы подготовки данных следующим образом: проанализировать текст любой программы, содержащей не менее 200—300 строк, и включить в таблицу все ключевые слова и идентификаторы программы.

10. Программа поиска по дереву

Составить программу формирования таблицы имен в виде дерева поиска и последующего поиска имени в таблице.

Оценить выигрыш по скорости поиска для оптимального дерева поиска (построенного по предварительно отсортированной и считанной с последовательными отступлениями влево и вправо от середины в последовательности имен) по сравнению с деревом, сформированным при произвольном порядке поступления входных имен.

В качестве аналога скорости поиска принять среднее число просмотренных при поиске элементов.

Оценить проигрыш в скорости построения дерева.

Файл с исходным списком символических имен сформировать с помощью подпрограммы подготовки данных следующим образом: проанализировать текст любой программы, содержащей не менее 200—300 строк, и включить в таблицу все ключевые слова и идентификаторы программы.

11. Программа поиска методом расстановки

Составить программу формирования таблицы символических имен методом расстановки с раздельным сцеплением и последующего поиска имени в таблице.

Сравнить по скорости поиска в таблице (в качестве аналога скорости поиска принять среднее число просмотренных при поиске элементов) различные варианты выбора функции расстановки, проанализировать 3—4 варианта.

Оценить коэффициент заполнения таблицы и среднюю длину поиска при занесении 25%, 50%, 75% и 100% имен из файла. Оптимизировать параметры функции расстановки по результатам этих исследований.

Файл с исходным списком символических имен сформировать с помощью подпрограммы подготовки данных следующим образом: проанализировать текст любой программы, содержащей не менее 200—300 строк, и включить в таблицу все ключевые слова и идентификаторы программы.

12. Программа нахождения максимального пути в ориентированном графе

Составить программу нахождения всех максимальных путей во взвешенном ориентированном графе с одним начальным и одним конечным узлом, используя представление графа с помощью связанного списка заголовочных узлов, каждый из которых является головой списка дуговых узлов.

Минимальный набор операций для структуры — графа должен включать следующие операции:

- формирование пустого графа;
- проверка наличия узлов в графе;
- проверка наличия заданного узла в графе;
- проверка наличия дуги между заданными узлами графа;
- включение нового узла в граф;
- исключение узла из графа;
- включение дуги с заданным весом между двумя заданными узлами (если дуга существует, то устанавливается нужный вес);
- исключение дуги между заданными узлами с выдачей веса дуги.

Дополнительно включаются все операции, необходимые для работы с графом при решении задачи.

Результаты работы программы должны быть представлены в виде списка узлов, принадлежащих максимальным путям, и значений соответствующих максимальных путей.

13. Программа нахождения минимальных расстояний между узлами графа

Составить программу нахождения минимальных расстояний между всеми или заданными узлами неориентированного взвешенного графа, используя представление графа с помощью связанного списка заголовочных узлов, каждый из которых является головой списка дуговых узлов.

Минимальный набор операций для структуры — графа должен включать следующие операции:

- формирование пустого графа;
- проверка наличия узлов в графе;
- проверка наличия заданного узла в графе;
- проверка наличия дуги между заданными узлами графа;
- включение нового узла в граф;
- исключение узла из графа;
- включение дуги с заданным весом между двумя заданными узлами (если дуга существует, то устанавливается нужный вес);
- исключение дуги между заданными узлами с выдачей веса дуги.

Дополнительно включаются все операции, необходимые для работы с графом при решении задачи.

Результаты работы программы должны быть представлены в виде списка вершин, принадлежащих минимальным путям, и значений соответствующих минимальных путей.

14. Программа планирования действий

Составить программу планирования действий при решении задачи, представленной ориентированным невзвешенным графом работ при наличии любого необходимого числа исполнителей этих работ, используя представление графа с помощью связанного списка заголовочных узлов, каждый из которых является головой списка дуговых узлов.

Результаты решения выдавать в виде перечня работ, которые необходимо выполнить в каждый очередной момент времени,

Спланировать действия при наличии только одного исполнителя, при этом выводится порядок решения работ (задача топологической сортировки).

Минимальный набор операций для структуры — графа должен включать следующие операции:

- формирование пустого графа;
- проверка наличия узлов в графе;
- проверка наличия заданного узла в графе;
- проверка наличия дуги между заданными узлами графа;
- включение нового узла в граф;
- исключение узла из графа;
- включение дуги с заданным весом между двумя узлами;
- исключение дуги между заданными узлами.

Дополнительно включаются все операции, необходимые для работы с графом при решении задачи.

15. Программа построения сетевого графика и определения характеристик его работ при неограниченном числе исполнителей

Составить программу нахождения основных характеристик сетевого графика выполнения некоторой задачи, представляющего собой ориентированный взвешенный граф, в котором каждая дуга представляет работу, вес дуги — время выполнения этой работы, каждый узел представляет время, к которому могут быть завершены работы, оканчивающиеся в этом узле. Использовать представление графа с помощью связанного списка заголовочных узлов, каждый из которых является головой списка дуговых узлов.

Характеристики сетевого графика:

- самые ранние моменты времени завершения работ, оканчивающихся в каждом узле;
- самые поздние моменты времени завершения работ, оканчивающихся в каждом узле;
- перечень узлов, у которых эти характеристики равны (критический путь).

Минимальный набор операций для структуры — графа должен включать следующие операции:

- формирование пустого графа;
- проверка наличия узлов в графе;
- проверка наличия заданного узла в графе;
- проверка наличия дуги между заданными узлами графа;

- включение нового узла в граф;
- исключение узла из графа;
- включение дуги с заданным весом между двумя заданными узлами (если дуга существует, то устанавливается нужный вес);
- исключение дуги между заданными узлами с выдачей веса дуги.

Дополнительно включаются все операции, необходимые для работы с графом при решении задачи.

16. Программа построения сетевого графика и определения характеристик его работ при ограниченном числе исполнителей

Составить программу нахождения основных характеристик сетевого графика выполнения некоторой задачи, представляющего собой ориентированный взвешенный граф, в котором каждая дуга представляет работу, вес дуги — время выполнения этой работы, каждый узел представляет время, к которому могут быть завершены работы, оканчивающиеся в этом узле. Использовать представление графа с помощью связанного списка заголовочных узлов, каждый из которых является головой списка дуговых узлов.

Характеристики сетевого графика:

- самые ранние моменты времени завершения работ, оканчивающихся в каждом узле;
- самые поздние моменты времени завершения работ, оканчивающихся в каждом узле;
- перечень узлов, у которых эти характеристики равны (критический путь).

Составить программу планирования действий и решения перечисленных выше задач при условии, что параллельно может выполняться не более m работ ($m \geq 1$).

Минимальный набор операций для структуры — графа должен включать следующие операции:

- формирование пустого графа;
- проверка наличия узлов в графе;
- проверка наличия заданного узла в графе;
- проверка наличия дуги между заданными узлами графа;
- включение нового узла в граф;
- исключение узла из графа;
- включение дуги с заданным весом между двумя заданными узлами (если дуга существует, то устанавливается нужный вес);
- исключение дуги между заданными узлами с выдачей веса дуги. Дополнительно включаются все операции, необходимые для работы с графом при решении задачи.

17. Библиотечный модуль для реализации одномерных динамических массивов

Реализовать библиотечный модуль для работы с одномерными массивами, память под которые выделяется динамически во время выполнения программы. В качестве базовой структуры для динамического массива использовать односвязный список.

Набор операций со списком:

- создание пустого списка;
- удаление списка из памяти;
- добавление элемента в начало списка.

Основной набор операций с одномерным динамическим массивом:

- создание пустого массива заданной длины;
- удаление массива из памяти;
- получение указателя на элемент с заданным индексом;
- запись значения в массив;
- чтение элемента массива;
- определение размера массива.

Реализовать матричные операции (копирование, сложение, скалярное произведение массивов, вывод и др.)

Продемонстрировать работу с динамическими массивами.

18. Библиотечный модуль для реализации двумерных динамических массивов

Реализовать библиотечный модуль для работы с двумерными массивами, память под которые выделяется динамически во время выполнения программы. В качестве базовой (родительской) структуры для динамического массива использовать односвязный список.

Набор операций со списком:

- создание пустого списка;
- удаление списка из памяти;
- добавление элемента в начало списка.

Основной набор операций с двумерным динамическим массивом:

- создание пустого массива заданной длины;
- удаление массива из памяти;
- получение указателя на элемент с заданным индексом;
- запись значения в массив;
- чтение элемента массива;
- определение размера массива.

Реализовать матричные операции (копирование, сложение, умножение, транспонирование, вывод и др.)

Продемонстрировать работу с динамическими массивами.

19. Программа построения дерева оптимального поиска

Составить программу формирования таблицы символических имен в виде дерева оптимального поиска и последующего поиска имени в таблице.

Файл с исходным списком символических имен сформировать с помощью подпрограммы подготовки данных следующим образом: проанализировать текст любой программы, содержащей не менее 200—300 строк, и включить в таблицу все ключевые слова и идентификаторы программы.

20. Реализация разреженного вектора

Реализовать объект — разреженный вектор на базе односвязного списка, каждый элемент которого содержит значение ненулевого элемента вектора и его номер. Элементы вектора — целые числа. Реализовать следующие операции с разреженным вектором: создание пустого вектора, чтение значения заданного элемента из вектора, запись значения заданного элемента в вектор, поиск минимального и максимального элементов.

Дополнительно предусмотреть и реализовать операции вывода сообщений о возможных ошибках при работе с разреженными векторами, а также векторные операции (копирование, сложение, скалярное умножение, умножение вектора на число и др.)

Продемонстрировать операции с созданным типом данных.

21. Реализация разреженной матрицы на базе списка ненулевых элементов

Реализовать объект — разреженную матрицу на базе односвязного списка, каждый элемент которого содержит значение ненулевого элемента матрицы, его номер строки и столбца. Элементы матрицы — целые числа. Реализовать следующие операции с разреженной матрицей: создание пустой матрицы, чтение значения заданного элемента из матрицы, запись значения заданного элемента в матрицу, поиск минимального и максимального элементов.

Дополнительно предусмотреть и реализовать операции вывода сообщений о возможных ошибках при работе с разреженными матрицами, а также матричные операции (копирование, сложение, умножение, транспонирование и др.)

22. Реализация разреженной матрицы на базе списка указателей

Реализовать объект — разреженную матрицу на базе списка указателей на списки ненулевых элементов строк. Каждый элемент списка указателей содержит номер строки и указатель на список ненулевых элементов этой строки. Каждый элемент списка ненулевых элементов содержит значение ненулевого элемента строки матрицы и его номер в строке (номер столбца). Элементы матрицы — целые числа.

Реализовать следующие операции с разреженной матрицей: создание пустой матрицы, чтение значения заданного элемента из матрицы, запись значения заданного элемента в матрицу, поиск минимального и максимального элементов.

Предусмотреть и реализовать операции вывода сообщений о возможных ошибках при работе с разреженными матрицами, а также матричные операции (копирование, сложение, умножение, транспонирование и др.)

23. Реализация множества строк на базе линейного списка строк

Реализовать объект — множество с базовым типом String на основе односвязного списка строк стандартного типа ShortString. Строки располагаются в множестве в порядке возрастания их значений. Реализовать все операции, определенные для стандартного типа множества в Delphi.

Продемонстрировать операции с созданным типом данных.

Проанализировать текст любой программы, содержащей 20—30 строк, и вывести зарезервированные слова программы.

24. Реализация множества строк на базе динамического массива строк

Реализовать объект — множество с базовым типом String на основе массива указателей на строки стандартного типа String, размещаемого в динамической памяти. Строки в множестве располагаются в порядке возрастания их значений. Реализовать все операции, определенные для стандартного типа множества в Delphi.

Продемонстрировать все операции с созданным типом данных.

Проанализировать текст любой программы, содержащей 20—30 строк, и вывести зарезервированные слова, не использованные в программе.

25. Реализация множества целых чисел на базе линейного списка

Реализовать объект — множество с базовым типом Integer на основе односвязного списка целых чисел. Числа располагаются в множестве в порядке возрастания их значений. Реализовать все операции, определенные для стандартного типа множества в Delphi.

Продемонстрировать все операции с созданным типом данных. Вывести все целые числа в заданном текстовом файле.

26. Реализация множества целых чисел на базе динамического массива

Реализовать объект — множество с базовым типом Integer на основе массива величин типа Integer, размещаемого в динамической памяти. Числа располагаются в множестве в порядке возрастания их значений.

Реализовать все операции, определенные для стандартного типа множества в Delphi.

Продемонстрировать все операции с созданным типом данных. Вывести все отрицательные целые числа в заданном текстовом файле.

27. Реализация множества вещественных чисел на базе линейного списка

Реализовать объект — множество с базовым типом Real на основе односвязного списка вещественных чисел. Числа располагаются в множестве в порядке возрастания их значений.

Реализовать все операции, определенные для стандартного типа множества в Delphi.

Продемонстрировать операции с созданным типом данных.

Вывести все числа в заданном текстовом файле.

28. Реализация множества вещественных чисел на базе динамического массива

Реализовать объект — множество с базовым типом Real на основе массива величин типа Real, размещаемого в динамической памяти. Числа располагаются в множестве в порядке возрастания их значений. Реализовать все операции, определенные для стандартного типа множества в Delphi.

Продемонстрировать все операции с созданным типом данных. Вывести все не целые числа в заданном текстовом файле.

29. Реализация длинных строк на базе циклического списка

Реализовать объект — длинная строка (>256 символов) на базе циклического списка стандартных строк. Реализовать все операции, определенные для стандартного типа строка в Delphi.

Продемонстрировать все операции с созданным типом данных. Проанализировать текст любой программы, содержащей 20 — 30 строк, сцепить в одну строку (отделяя пробелом) все идентификаторы программы (без повторений) и вывести полученную строку в файл результатов.

30. Программа сортировки данных во внешнем файле

Составить программу сортировки данных, содержащихся в текстовом файле последовательного доступа, методом l лент (l — четное число). Тип и числовые характеристики данных в исходном файле выбрать произвольно.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ

«Зачтено» — Если в отчете РАР 1 излагается алгоритм решения поставленной задачи, приводится соответствующий фрагмент программы, описываются данные, используемые для проверки метода решения задачи, — структура данных и место их хранения.

«Не зачтено» - если в работе отсутствует алгоритм решения поставленной задачи, не приводится соответствующий фрагмент программы, не описываются данные, используемые для проверки метода решения задачи, — структура данных и место их хранения.

3.1.3 Выполнение и сдача РАР 2

3.1.3.1 Место РАР 2 в структуре дисциплины

Разделы дисциплины, освоение которых обучающимися сопровождается или завершается выполнением РАР 2		Компетенции, формирование/развитие которых обеспечивается в ходе выполнения РАР 2
№	Наименование	
15	Уникальные возможности языка программирования Java, Python	ОПК - 2

3.1.3.2 Перечень примерных тем РАР 2

Задание 1 для РАР 2

Исходные данные необходимо оформить в виде массива. При выполнении задания ввод исходных данных и вывод результатов сопровождать комментариями (какие данные нужно ввести и что получается в результате). Составить блок-схему программы. Оформить отчет.

1. Подсчитать среднемесячную зарплату сотрудника предприятия.
2. Дан объем продукции, выпущенной заводом за год (помесячно). Найти наименьший объем. В качестве результата вывести номер месяца и объем выпущенной продукции.
3. Курс доллара в течение года менялся в диапазоне от 28руб. до 30руб. Найти наибольшее значение курса доллара. В качестве результата вывести номер месяца и значение курса доллара.
4. Известен месячный план выпуска некоторой продукции и объемы выпущенной этой продукции заводом за год (помесячно). Определить, когда завод не выполнил план. Результат получить в виде: номер месяца и объем выпущенной продукции.
5. Даны результаты сдачи экзамена по информатике группы студентов (в группе 20 студентов). Подсчитать количество студентов, не сдавших экзамен.
6. Известна среднемесячная зарплата 10 сотрудников отдела. Расположить данные в порядке убывания.
7. Известен годовой процент на вклад с капитализацией (начисление процентов ежемесячно). Определить, сколько денег получит вкладчик в конце года, если на 1 января сумма вклада составляла 1500руб. в качестве результата вывести сумму вклада на конец каждого месяца.
8. Известны данные по продаже компьютеров в течение недели. Найти общее количество проданных компьютеров.
9. Известны данные по продаже компьютеров в течение недели. Расположить эти данные в порядке возрастания.
10. Известен месячный план выпуска некоторой продукции и объемы выпущенной продукции заводом за год (помесячно). Определить месяц, в котором было максимальное отклонение от плана. В качестве результата вывести номер месяца и отклонение.
11. Известен месячный план выпуска некоторой продукции и объемы выпущенной продукции заводом за год (помесячно). Определить, был ли выполнен годовой план.
12. Даны результаты сдачи экзамена по информатике группы студентов (в группе 20 студентов). Подсчитать количество студентов, сдавших экзамен без троек.
13. Известен месячный план выпуска некоторой продукции и объемы выпущенной этой продукции заводом за год (помесячно). Определить, когда завод перевыполнил план. Результат получить в виде: номер месяца и объем продукции, выпущенной сверх плана.
14. Подсчитать среднемесячную зарплату сотрудника предприятия и найти зарплату, которая наиболее близка к средней. В качестве результата вывести среднюю зарплату, наиболее близкую и ее номер в массиве.
15. Даны результаты сдачи экзамена по информатике группы из 15 студентов. Подсчитать количество студентов, не сдавших экзамен, в численном и в процентном соотношении.

Задание 2 для РАР 2

Исходные данные необходимо оформить в виде двумерного массива, в части заданий использовать дополнительно и одномерные массивы. При выполнении задания ввод исходных данных и вывод результатов сопровождать комментариями (какие данные нужно ввести и что получается в результате).

1. Известен план выпуска компьютеров и количество выпущенных компьютеров тремя фирмами за шесть месяцев. Определить для каждой фирмы, был ли выполнен план по итогам шести месяцев.
2. Известно количество сделанных столов тремя фабриками за два квартала. Определить максимальное количество выпущенных столов. В качестве результата вывести месяц, в котором это было, и название фабрики.
3. Известен план выпуска компьютеров и количество выпущенных компьютеров тремя фирмами за три месяца. Определить, в каком месяце не был выполнен план третьей фирмой.
4. Известен план выпуска компьютеров и количество выпущенных компьютеров тремя фирмами за шесть месяцев. Определить для каждой фирмы количество месяцев, когда план был перевыполнен.
5. Известна заработная плата, полученная 5 сотрудниками отдела в течение года. Определить максимальную заработную плату. В качестве результата вывести фамилию и размер заработной платы.
6. Известно количество выпущенной продукции тремя заводами за первый квартал (помесячно). Найти среднемесячное количество выпущенной продукции для каждого завода.
7. Известно количество выпущенной продукции тремя заводами за первый квартал (помесячно). Найти среднемесячное количество выпущенной продукции по всем заводам.
8. Известны результаты сдачи трех экзаменов пятью студентами. Найти фамилии студентов, не сдавших оба экзамена.
9. Известно количество сделанных столов тремя фабриками за два квартала. Определить, какая фабрика выпустила максимальное количество столов по итогам шести месяцев.
10. Известна заработная плата, полученная 10 сотрудниками отдела в течение года. Определить среднемесячную зарплату по отделу.
11. Известны результаты сдачи двух экзаменов семью студентами. Найти фамилии студентов, не сдавших хотя бы один экзамен.
12. Известно количество сделанных столов тремя фабриками за два квартала. Определить, какая фабрика выпустила минимальное количество столов по итогам первых трех месяцев.
13. Известны результаты сдачи трех экзаменов десятью студентами. Найти средний балл каждого студента и общий средний балл. Точность среднего балла – два знака после запятой.
14. Известно количество сделанных столов тремя фабриками за два квартала. Определить, какая фабрика выпустила максимальное количество столов по итогам второго квартала.
15. Известны результаты сдачи двух экзаменов десятью студентами. Определить фамилии студентов, сдавших экзамены без троек.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ

«Зачтено» – Если в отчете РАР 2 излагается алгоритм решения поставленной задачи, приводится соответствующий фрагмент программы, описываются данные, используемые для проверки метода решения задачи, — структура данных и место их хранения.

«Не зачтено» - если в работе отсутствует алгоритм решения поставленной задачи, не приводится соответствующий фрагмент программы, не описываются данные, используемые для проверки метода решения задачи, — структура данных и место их хранения.

3.1.2. ВОПРОСЫ для проведения входного контроля

1. В чем сущность концепции типа данных

- a) Тип характеризует множество значений, к которому принадлежит константа, которое может принимать переменная или выражение или может обрабатывать функция.
- b) компьютер может использовать информацию о типах для проверки вычислимости и правильности различных конструкций.
- c) Тип характеризует внешний вид константы, переменной или выражения или функции.
- d) возможности создания других типов на основе базовых.

2. Методами структурирования называются...

- a) правила, по которым из простых типов можно строить структурированные типы любой сложности.
- b) правила, по которым проводятся вычисления.
- c) правила, по которым описываются типы, создаваемые пользователем.
- d) правила, по которым определяется множество операций над данными указанного типа.

3. Данное характеризуется

- a) типом и значением.
- b) именем и значением.
- c) именем, типом и значением.

4. Константой называют...

- a) данное, значение которого нельзя изменить в ходе выполнения программы.
- b) данное, имя которого нельзя изменить в ходе выполнения программы.
- c) данное, тип которого нельзя изменить в ходе выполнения программы.

5. Переменная это...

- a) данное, значение которого можно определить только в ходе выполнения программы
- b) данное, значение которого нельзя определить в ходе выполнения программы
- c) данное, тип которого можно определить только в ходе выполнения программы

6. Данное типа Boolean может принимать следующие значения

- a) 0, 1
- b) true и false
- c) true и nil

7. Значением символьного данного (char) может быть...

- a) один символ из множества печатанных символов
- b) множество печатанных символов
- c) множество целых чисел
- d) цифра

1. Типы byte (байтовый) и shorting (короткий целый) отличаются...

- a) диапазоном
- b) объемом занимаемой памяти
- c) диапазоном и объемом занимаемой памяти

9. Типы single и real отличаются...

- a) диапазоном
- b) объемом занимаемой памяти
- c) диапазоном и объемом занимаемой памяти

10. type T = (C1, C2, ... Cn); описанный таким образом тип называют...

- a) ограниченным
- b) множественным
- c) перечисляемым

11. Укажите номер верного описания типа

- a) Type буква = 'a'..'я';
- b) Type буква = a..я;
- c) Type буква = 'a'...'я';

ШКАЛА И КРИТЕРИИ ОЦЕНКИ
ответов на вопросы входного контроля

Шкала и критерии оценивания для аудиторных письменных тестирований	
Отлично	Более 90% тестовых заданий решены верно
Хорошо	От 70 до 89% тестовых заданий решены верно
Удовлетворительно	От 60 до 69% тестовых заданий решены верно
Неудовлетворительно	Менее 60% тестовых заданий решены верно

3.1.3 Средства для текущего контроля

ВОПРОСЫ

для самостоятельного изучения темы

«Программирование вычислительных процессов циклической структуры на основе операторов цикла с параметром, предусловием и постусловием. Циклические алгоритмы, их типы»

- 1) Оператор цикла с параметром
- 2) Оператор цикла с постусловием
- 3) Оператор цикла с предусловием
- 4) Вложенные циклы
- 5) Оператор прерывания цикла

ВОПРОСЫ

для самостоятельного изучения темы

«Массивы: понятие, основные характеристики, элементы. Обработка массивов данных»

- 1) Структурные типы данных. Способы представления в памяти.
- 2) Описание статических массивов в программе. Доступ к элементам.
- 3) Массивы констант. Ввод-вывод элементов двумерного массива.
- 4) Нахождения максимального элемента массива и его номера.
- 5) Упорядочивание массива по возрастанию. Поиск в массиве.
- 6) Динамические массивы.

ВОПРОСЫ

для самостоятельного изучения темы

«Структурированные типы данных (массив, строка, запись, множество, файл). Организация ветвлений Строки»

- 1) Структурированные типы данных (массивы, файлы, записи, множества).
- 2) Динамические структуры данных. Динамическая память. Управление памятью.
- 3) Управление файлами. Страничная память. Стратегии управления страничной памятью
- 4) Алгоритм fifo – Выталкивание первой пришедшей страницы – Простейший алгоритм.
- 5) Операции над файлами. Организация ввода-вывода. Жизненный цикл программ.

ВОПРОСЫ

для самостоятельного изучения темы

«Двумерные массивы»

- 1) Двумерные массивы: как с ними работать.
- 2) Назначение двумерных массивов.
- 3) Понятие массива. Статический массив. Динамический массив. Строка как массив

ВОПРОСЫ

для самостоятельного изучения темы

«Средства обработки текстовых и типизированных файлов»

- 1) Общие принципы работы. Обработка ошибок. Текстовые файлы
- 2) Типизированные файлы.
- 3) Нетипизированные файлы. Прочее

ВОПРОСЫ

для самостоятельного изучения темы

«Графика в Pascal ABC.NET»

- 1) Управление графическим окном. Процедуры рисования графических примитивов.
- 2) Процедуры, используемые для работы с цветом. Цвета в PascalABC
- 3) Процедуры для работы с текстом.

ВОПРОСЫ

для самостоятельного изучения темы

«Графика. Работа с изображениями»

- 1) Технологии работы с графической информацией. Виды графической информации

- 2) Электронные таблицы. Назначение и основные возможности.
- 3) Средства обработки графической информации.
- 4) 5 популярных программ для обработки графической информации

ВОПРОСЫ

для самостоятельного изучения темы

«Язык программирования Java. Уникальные возможности языка программирования Java»

- 1) Преимущества языка программирования Java. Востребованность, рейтинги, области применения
- 2) Технические особенности и достоинства
- 3) Краткая теория: особенности и сферы применения Java. Что пишут на языке Java: сферы применения. Какие сайты и программы работают на Java
- 4) Что гибче, быстрее и легче: сравниваем Java, Python и C

ОБЩИЙ АЛГОРИТМ

самостоятельного изучения темы

1) Ознакомиться с рекомендованной учебной литературой и электронными ресурсами по теме (ориентируясь на вопросы для самоконтроля).
2) На этой основе составить развёрнутый план изложения темы
3) Выбрать форму отчетности конспектов(план – конспект, текстуальный конспект, свободный конспект, конспект – схема)
2) Оформить отчётный материал в установленной форме в соответствии методическими рекомендациями
3) Провести самоконтроль освоения темы по вопросам, выданным преподавателем
4) Предоставить отчётный материал преподавателю по согласованию с ведущим преподавателем
5) Подготовиться к предусмотренному контрольно-оценочному мероприятию по результатам самостоятельного изучения темы
6) Принять участие в указанном мероприятии, пройти рубежное тестирование по разделу на аудиторном занятии и заключительное тестирование в установленное для внеаудиторной работы время

ШКАЛА И КРИТЕРИИ ОЦЕНКИ

самостоятельного изучения темы

Шкала и критерии оценивания для аудиторных письменных тестирований	
Отлично	Более 90% тестовых заданий решены верно
Хорошо	От 70 до 89% тестовых заданий решены верно
Удовлетворительно	От 60 до 69% тестовых заданий решены верно
Неудовлетворительно	Менее 60% тестовых заданий решены верно

ВОПРОСЫ

для самоподготовки к практическим (семинарским) занятиям

ВОПРОСЫ

для самоподготовки к практическим занятиям

1. Объектно-ориентированное программирование. Основные понятия: объект, класс, инкапсуляция, полиморфизм, наследование. Создание класса и объекта в программе. Реализация методов класса. Конструкторы и деструкторы.
2. Массивы объектов. Указатели на объекты. Доступ к членам класса с помощью объекта и указателя.
3. Наследование объектов. Создание класса и объекта потомка на базе родительского класса и объекта. Вызов родительских методов. Виртуальные и переопределяемые методы.
4. Динамическая память: применение, особенности обработки. Указатели и динамическая память. Управление динамической памятью: выделение и освобождение, запись необходимой информации.
5. Динамический массив, его назначение, использование, сравнение со статическим массивом. Создание динамического массива и работа с ним. Доступ к элементу динамического массива.
6. Односвязный и двусвязный динамический список, их назначение и применение. Алгоритмы построения списков. Основные операции над динамическими списками.

7. Понятие и назначение файлов. Типы файлов. Открытие файла, чтение и запись информации в файл, закрытие файла, проверка достижения конца файла. Примеры фрагментов программного кода обработки текстовых файлов.
8. Двоичные файлы: применение и особенности. Основные операции с двоичными файлами. Прямой доступ к файлу. Блочная обработка двоичных файлов.
9. Понятие, назначение структур. Примеры применения структур. Создание структурного шаблона и описание структурных переменных. Доступ к полям структуры.
10. Массивы структур. Основные операции с массивом структур. Создание и заполнение массива структур информацией. Моделирование работы с базой данных с применением массива структур.
11. Структурированные типы данных: двумерные массивы. Создание и заполнение массива информацией. Доступ к элементу массива. Основные операции с двумерным массивом. Передача двумерных массивов функциям.
12. Обработка отдельных строк и столбцов таблицы.
13. Квадратная матрица и ее разновидности. Операции с квадратной матрицей. Матричные операции.
14. Структурированные типы данных: одномерные массивы. Создание и заполнение массива информацией. Доступ к элементу массива. Основные операции с одномерным массивом.
15. Различные алгоритмы сортировок одномерного массива. Реализация алгоритмов на языке программирования C/C++. Сравнение эффективности алгоритмов сортировки.
16. Виды поиска элемента в одномерном массиве: линейный и бинарный поиск. Реализация алгоритмов на языке программирования C/C++. Сравнение эффективности алгоритмов поиска.
17. Логические выражения. Операции математической логики. Команда if, правила ее использования. Неполная, полная и вложенная ветвь.
18. Команда switch, правила и ограничения ее использования. Применение команды switch в практических задачах.

Задачи для практических занятий

На практических занятиях студенты решают задачи по теоретическим основам программирования. Задачи выдаются преподавателем на каждое занятие.

ШКАЛА И КРИТЕРИИ ОЦЕНКИ самоподготовки по темам практических занятий

- *оценка «отлично»* выставляется обучающемуся, если студент на все вопросы дает аргументированные ответы и показывает теоретические знания по вопросам, выносимым на практические занятия. Задачи решаются верно.
- *оценка «хорошо»* - выставляется обучающемуся, если студент показывает теоретические знания по вопросам, выносимым на практические занятия, но не на все вопросы дает аргументированные ответы. Задачи решаются верно, или с некоторыми трудностями.
- *оценка «удовлетворительно»* - выставляется обучающемуся, если при ответе на вопросы практических занятий студент допускает ошибки. При решении задач студент допускает некоторые ошибки.
- *оценка «неудовлетворительно»* - выставляется обучающемуся, если наблюдается частичное или полное не владение темой практического занятия, студент не дает правильные ответы на заданные вопросы. При решении задач студент допускает ошибки.

Текущий контроль по теме «Основы программирования в системе, интегрированной в пакет офисных программ»

Вариант 1

1. Системы для разработки новых программ на конкретном языке программирования:
 - а) системы программирования +
 - б) программированные системы
 - в) системы ретуширования
2. К какому уровню языков относятся языки ассемблера:
 - а) среднего уровня
 - б) низкого уровня +
 - в) высокого уровня
3. Какой язык программирования был создан в 1979 году и назван в честь первого в мире программиста:
 - а) Паскаль

- б) Фортран
- в) Ада +

4. Для чего используют программы-ассемблеры:

- а) для перевода программ в машинные коды +
- б) для обеспечения бесперебойной работы программ
- в) для создания программ

5. Какой язык называется машинно-ориентированным:

- а) язык, в основу которого заложены принципы объектно-ориентированного программирования
- б) любой универсальный язык программирования
- в) язык, определяющийся набором команд конкретного процессора +

6. Какой язык программирования, созданный в 1957 году, является одним из первых алгоритмических языков и до сих пор применяется для научных вычислений:

- а) Паскаль
- б) Фортран +
- в) Ада

7. Что такое системы программирования:

- а) программные средства для обеспечения бесперебойной работы существующих программ
- б) программные средства для перевода команд с естественного языка в машинные коды
- в) программные средства для создания и отладки новых программ +

8. Выберите верное утверждение о языке ассемблер:

- а) программа, написанная на Ассемблере для одного процессора не будет работать на другом +
- б) программы, написанные на языке Ассемблер создаются только в среде Linux
- в) программа, написанная на Ассемблере для одного процессора будет работать на любом другом

9. Как называются формальные языки, созданные для разработки программ:

- а) языки высокого уровня
- б) популярные языки
- в) алгоритмические языки +

10. Как называют программы, предназначенные для перевода в машинные коды программы, написанной на языке высокого уровня:

- а) транслитеры
- б) трансляторы +
- в) конденсаторы

11. Из слов какого языка строятся команды языков программирования высокого уровня:

- а) естественного языка +
- б) логического языка
- в) алгоритмического языка

12. Одно из самых удобных средств разработки программ современных систем программирования:

- а) ретранслятор
- б) компилятор или интерпретатор +
- в) интерстеллер

13. Одно из самых удобных средств разработки программ современных систем программирования:

- а) интегрированная среда разработки +
- б) интригованная среда разработки
- в) интегрирующая среда разработки

14. Одно из самых удобных средств разработки программ современных систем программирования:

- а) тихий режим работы
- б) однооконный режим работы
- в) многооконный режим работы +

15. Одно из самых удобных средств разработки программ современных систем программирования:

- а) встроенный ассемблер +
- б) встроенный кассемблер

в) встроенный ассемблятор

16. Одна из популярных систем программирования:

- а) Basic C
- б) Turbo Basic +
- в) Basic S

17. Одна из популярных систем программирования:

- а) Slow Basic
- б) Fast Basic
- в) Quick Basic +

18. Одна из популярных систем программирования:

- а) Fast Pascal
- б) Turbo Pascal +
- в) Slow Pascal

19. Одна из популярных систем программирования:

- а) Turbo W
- б) Turbo S
- в) Turbo C +

20. В последнее время получили распространение системы программирования, ориентированные на создание:

- а) Yandex-приложений
- б) Windows-приложений +
- в) Google-приложений

21. Язык Паскаль был разработан в этом году:

- а) 1970 +
- б) 1980
- в) 1990

22. Язык Си разработан Деннисом Ритчи в этом году:

- а) 1982
- б) 1972 +
- в) 1985

23. Один из программных компонентов системы программирования:

- а) галерея подпрограмм
- б) регулятор текста
- в) редактор текста +

24. Один из программных компонентов системы программирования:

- а) ретранслятор с соответствующего языка
- б) транслятор с соответствующего языка +
- в) регулятор соответствующего языка

25. Один из программных компонентов системы программирования:

- а) наладчик
- б) постановщик
- в) компоновщик (редактор связей) +

26. Один из программных компонентов системы программирования:

- а) доводчик
- б) отладчик +
- в) наладчик

27. Один из программных компонентов системы программирования:

- а) библиотеки подпрограмм +
- б) библиотеки программ
- в) галерея подпрограмм

28. Программа для ввода и модификации текста:

- а) компоновщик
- б) транслятор
- в) редактор текста +

29. Трансляторы делятся на столько классов:

- а) 2 +
- б) 3
- в) 4

30. Позволяет управлять процессом исполнения программы, является инструментом для поиска и исправления ошибок в программе:

- а) компоновщик
- б) отладчик +
- в) загрузчик

Вариант 2

1. Файл созданный в MS PowerPoint имеет расширение:

- а) .PPT +
- б) .XLS
- в) .DOCX

2. Каких списков нет в текстовом редакторе:

- а) маркированных
- б) точечных +
- в) нумерованных

3. Файл созданный в MS Excel имеет расширение:

- а) .DOC
- б) .PPT
- в) .XLS +

4. Правильное обозначение адреса ячейки в MS Excel:

- а) \$A1 +
- б) X
- в) G

5. Способы отображения таблиц БД в Access:

- а) Режим программа
- б) Режим цифры
- в) Режим Конструктор +

6. Правильное обозначение адреса ячейки в MS Excel:

- а) P
- б) C3 +
- в) O

7. Способы отображения таблиц БД в Access:

- а) Режим программа
- б) Режим строка
- в) Режим таблица +

8. Полосы прокрутки MS Word:

- а) позволяют перемещаться по листкам документов в горизонтальном и вертикальном направлениях +
- б) служат для установки полей страницы, отступов абзаца, позиций табуляции и размеров ячеек в таблицах
- в) содержат кнопки для быстрого вызова наиболее употребительных команд

9. Запросы MS Access:

- а) помогает вводить, просматривать и модифицировать информацию в таблице или запросе
- б) объект БД, предназначенный для печати данных
- в) объект БД, позволяющий получить нужные данные из одной или нескольких таблиц +

10. Отчеты MS Access:

- а) помогает вводить, просматривать и модифицировать информацию в таблице или запросе
- б) объект БД, предназначенный для печати данных +
- в) объект БД, позволяющий получить нужные данные из одной или нескольких таблиц

11. Ячейка MS Excel:

- а) основная единица хранения данных, образуемая на пересечении строки и столбца +
- б) состоит из отдельных рабочих листов, каждый из которых может содержать данные
- в) служит для организации и анализа данных (построение расчетных таблиц, диаграмм, работа с базами данных и т.д.)

12. Таблица MS Access:

- а) объект БД, предназначенный для печати данных
- б) это объект, предназначенный для хранения данных в виде записей (строк) и полей (столбцов). Обычно каждая таблица используется для хранения сведений по одному конкретному вопросу +
- в) объект БД, позволяющий получить нужные данные из одной или нескольких таблиц

13. Лист MS Excel:

- а) основная единица хранения данных, образуемая на пересечении строки и столбца
- б) состоит из отдельных рабочих листов, каждый из которых может содержать данные
- в) служит для организации и анализа данных (построение расчетных таблиц, диаграмм, работа с базами данных и т.д.) +

14. Слайды могут содержать:

- а) рисунки
- б) видеоклипы
- в) оба варианта верны +
- г) нет верного ответа

15. Панели инструментов MS Word:

- а) содержат кнопки для быстрого вызова наиболее употребительных команд +
- б) позволяют перемещаться по листкам документов в горизонтальном и вертикальном направлениях
- в) служат для установки полей страницы, отступов абзаца, позиций табуляции и размеров ячеек в таблицах

16. Слайды могут содержать:

- а) звуковые файлы
- б) диаграммы
- в) оба варианта верны +
- г) нет верного ответа

17. Строка меню MS Word:

- а) позволяют перемещаться по листкам документов в горизонтальном и вертикальном направлениях
- б) содержат кнопки для быстрого вызова наиболее употребительных команд
- в) Word содержит девять пунктов меню, в которых тематически сгруппированы все команды и инструменты, имеющиеся в распоряжении пользователя +

18. Файл созданный в MS WORD имеет расширение:

- а) .PPT
- б) .DOCX +
- в) .XLS

19. Пакет программ Microsoft Office включает:

- а) Excel
- б) Word
- в) оба варианта верны +
- г) нет верного ответа

20. Файл созданный в MS WORD имеет расширение:

- а) .XLS
- б) .DOC +
- в) .PPT

21. Пакет программ Microsoft Office включает:

- а) Access
- б) Power Point
- в) оба варианта верны +
- г) нет верного ответа

22. Что такое MS Excel:

- а) разработка и управление персональными базами данных
- б) выполнение расчетов, построение диаграмм, анализ данных, автоматизация трудоемких задач и прочие операции с электронными таблицами +
- в) многофункциональный инструмент для подготовки и оформления текстовых документов

23. Что такое MS PowerPoint:

- а) выполнение расчетов, построение диаграмм, анализ данных, автоматизация трудоемких задач и прочие операции с электронными таблицами
- б) многофункциональный инструмент для подготовки и оформления текстовых документов
- в) подготовка красочных и интерактивных презентаций как для семейных праздников, так и для бизнес-выступлений +

24. Книга MS Excel:

- а) служит для организации и анализа данных (построение расчетных таблиц, диаграмм, работа с базами данных и т.д.)
- б) состоит из отдельных рабочих листов, каждый из которых может содержать данные +
- в) основная единица хранения данных, образуемая на пересечении строки и столбца

25. Линейки форматирования MS Word:

- а) содержат кнопки для быстрого вызова наиболее употребительных команд
- б) позволяют перемещаться по листкам документов в горизонтальном и вертикальном направлениях
- в) служат для установки полей страницы, отступов абзаца, позиций табуляции и размеров ячеек в таблицах +

26. Что такое MS Access:

- а) многофункциональный инструмент для подготовки и оформления текстовых документов
- б) разработка и управление персональными базами данных +
- в) подготовка красочных и интерактивных презентаций как для семейных праздников, так и для бизнес-выступлений

27. Что такое MS Word:

- а) многофункциональный инструмент для подготовки и оформления текстовых документов +
- б) подготовка красочных и интерактивных презентаций как для семейных праздников, так и для бизнес-выступлений
- в) разработка и управление персональными базами данных

28. Объекты MS Access:

- а) таблицы
- б) формы
- в) оба варианта верны +
- г) нет верного ответа

29. Формы MS Access:

- а) помогает вводить, просматривать и модифицировать информацию в таблице или запросе +
- б) объект БД, предназначенный для печати данных
- в) объект БД, позволяющий получить нужные данные из одной или нескольких таблиц

30. Объекты MS Access:

- а) запросы
- б) отчеты
- в) оба варианта верны +
- г) нет верного ответа

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ

ответов на вопросы текущего контроля

Шкала и критерии оценивания для аудиторных письменных тестирований	
Отлично	Более 90% тестовых заданий решены верно
Хорошо	От 70 до 89% тестовых заданий решены верно
Удовлетворительно	От 60 до 69% тестовых заданий решены верно
Неудовлетворительно	Менее 60% тестовых заданий решены верно

3.1.4. Средства для промежуточной аттестации по итогам изучения дисциплины

Тестовые задания для прохождения итогового тестирования

Тестирование осуществляется по всем темам и разделам дисциплины, включая темы, выносимые на самостоятельное изучение. Процедура тестирования ограничена во времени и предполагает максимальное сосредоточение обучающегося на выполнении теста, содержащего несколько тестовых заданий.

Тестирование проводится письменной форме (на бумажном носителе). Тест включает в себя 30 вопросов. Время, отводимое на выполнение теста - 30 минут. В каждый вариант теста включаются вопросы разных типов (одиночный и множественный выбор, открытые (ввод ответа с клавиатуры), на упорядочение, соответствие и др.). На тестирование выносятся вопросы из каждого раздела дисциплины.

Подготовка к заключительному тестированию по итогам изучения дисциплины.

Тестирование осуществляется по всем темам и разделам дисциплины, включая темы, выносимые на самостоятельное изучение.

Процедура тестирования ограничена во времени и предполагает максимальное сосредоточение студента на выполнении теста, содержащего несколько тестовых заданий.

Студенту рекомендуется:

1) при неуверенности в ответе на конкретное тестовое задание пропустить его и переходить к следующему, не затрачивая много времени на обдумывание тестовых заданий при первом проходе по списку теста;

2) при распределении общего времени тестирования учитывать (в случае компьютерного тестирования), что в автоматизированной системе могут возникать небольшие задержки при переключении тестовых заданий.

Необходимо помнить, что:

1) тест является индивидуальным, общее время тестирования и количество тестовых заданий ограничены и определяются преподавателем в начале тестирования;

2) по истечении времени, отведённого на прохождение теста, сеанс тестирования завершается;

3) допускается во время тестирования только однократное тестирование;

4) вопросы студентов к преподавателю по содержанию тестовых заданий и не относящиеся к процедуре тестирования не допускаются.

Тестируемому во время тестирования запрещается:

1) нарушать дисциплину;

2) пользоваться учебно-методической и другой вспомогательной литературой, электронными средствами (мобильными телефонами, электронными записными книжками и пр.);

3) использование вспомогательных средств и средств связи на тестировании допускается при разрешении преподавателя-предметника;

4) копировать тестовые задания на съёмный носитель информации или передавать их по электронной почте;

5) фотографировать задания с помощью цифровой фотокамеры;

6) выносить из класса записи, сделанные во время тестирования.

Тестируемый имеет право:

1) вносить замечания о процедуре проведения тестирования и качестве тестовых заданий;

2) перенести сроки тестирования (по уважительной причине) по согласованию с преподавателем.

На рабочее место тестируемому разрешается взять ручку, черновик, калькулятор.

За несоблюдение вышеперечисленных требований преподаватель имеет право удалить тестируемого, при этом результат тестирования удаленного лица аннулируется.

Бланк теста
Образец
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Омский государственный аграрный университет имени П.А. Столыпина»

Тестирование по итогам освоения дисциплины «Теоретические основы программирования»
Для обучающихся направления подготовки 09.04.02 – Информационные системы и технологии

ФИО _____ группа _____

Дата _____

Уважаемые обучающиеся!

Прежде чем приступить к выполнению заданий внимательно ознакомьтесь с инструкцией:

1. Отвечая на вопрос с выбором правильного ответа, правильный, на ваш взгляд, ответ (ответы) обведите в кружок.
 2. В заданиях открытой формы впишите ответ в пропуск.
 3. В заданиях на соответствие заполните таблицу.
 4. В заданиях на правильную последовательность впишите порядковый номер в квадрат.
 4. Время на выполнение теста – 30 минут
 5. За каждый верный ответ Вы получаете 1 балл, за неверный – 0 баллов. Максимальное количество полученных баллов 30.
- Желаем удачи!

Вариант № 1

1. Что такое препроцессор?

Укажите **не менее одного** варианта ответа.

- Составная системного блока, предназначенная для обработки данных
- Составная процессора, предназначенной для вычислений с плавающей запятой
- + Составляющая компиляции, которая обрабатывает директивы или команды

2. С какого символа начинается запись директивы препроцессора?

Укажите **не менее одного** варианта ответа.

- @
- + #
- <
- »

3. Укажите правильные записи параметров директив:

Укажите **не менее двух** вариантов ответа.

- + <Math.h>
- «Math.h»
- + «D: \stud \mybib.h»
- <D: \stud \mybib.h>

4. Что такое заголовочные файлы?

Укажите **не менее двух** вариантов ответа.

- Название программы, указывается при сохранении
- Название главной функции или функции пользователя
- + Модули, сохраняют заголовки функций
- + Стандартные библиотеки, расположенные в папке include

5. Укажите команды, в результате выполнения которых после вывода данных осуществляется переход на следующую строку

Укажите **не менее двух** вариантов ответа.

- + Cout << «Ocinka» << 12 << endl
- Cout << endl << «Ocinka» << 12
- + Cout << «Ocinka» << 12 << «\ n»
- Cout << «Ocinka \ n» << 12.

6. Укажите, что будет выведено на экран в результате выполнения команды cout << «rez:» << setw (3) << 1234

Укажите **не менее одного** варианта ответа.

— 3
— 34
— 123
+ 1234

7. Выберите правильное описание переменных: переменная x для сохранения действительного типа, переменная a для сохранения целого числа, символьная переменная c; переменную c инициализировать значением 'n', переменную x инициализировать значением 18.21

— Float x, a; char c; c = 'n';
— Float x = 18.21; a; char c = 'n';
+ Float x = 18.21; int a; char c = 'n';
+ Float x; x = 18.21; int a; char c = 'n';

8. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного варианта ответа**.

```
int a, b
float c
a = 3
c = 2.6
b = a * c
— 6
+ 7
— 7.2
— 9
```

9. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного варианта ответа**.

```
int a, b
float c
a = 3
c = 3.5
b = a * (int) c
+ 9
— 10
— 11
— 12
```

10. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного варианта ответа**.

```
int a, b
float c
a = 5
c = 3.3
b = (int) (a * c)
— 15
+ 16
— 17
— 20
```

11. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного варианта ответа**.

```
int a, b
a = 5
b = 3.5 * ++a
— 15
— 17
— 18
+ 21
```

12. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного варианта ответа**.

```
int a, b
a = 2
b = 4.5 / (a++)
— 1
— 1.5
+ 2
— 2.25
```

13. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного** варианта ответа.

```
int a, b
```

```
a = 4
```

```
b = 35 / — — a
```

— 8

— 9

+ 11

— 12

14. Определите значение переменной b после выполнения следующих команд:

Укажите **не менее одного** варианта ответа.

```
int a, b
```

```
a = 2
```

```
b = 4.5 * (a - -)
```

— 4

— 5

— 8

+ 9

15. Определите значение переменной x после выполнения следующих команд:

Укажите **не менее одного** варианта ответа.

```
int a = — 2, b = 3, x
```

```
x = abs (a) + pow (b, 3) / 2
```

— 5

— 6

— 14

+ 15

16. Выберите составные части определения массива:

Укажите **не менее трех** вариантов ответа.

+ Совокупность переменных одного типа

+ Переменные хранятся в последовательно расположенных ячейках оперативной памяти

+ Последовательность переменных, которые имеют одинаковое имя

— Благодаря нумерации переменные располагаются только в строку

17. Укажите, что может быть индексом элемента массива

Укажите **не менее двух** вариантов ответа.

— Символ

— Дробное число

+ Целое положительное число

+ Математическое выражение, результат вычисления которого является целое положительное число

18. Выберите верные утверждения, имеющие отношение к имени массива:

Укажите **не менее двух** вариантов ответа.

— Название массива — ключевое слово `array` и любая латинская буква

+ Название массива является указателем на его первый элемент

— Название массива является указателем на его элемент с индексом [1]

+ Имя массива создается по правилам создания идентификатора

19. Выберите верные утверждения, имеющие отношение к размеру массива:

Укажите **не менее одного** варианта ответа.

+ Размером могут быть переменные целого типа

— Размером могут быть переменные вещественного типа

— Размер — это индекс последнего элемента массива

20. Укажите описания массивов, которые нельзя использовать ни в какой части программы:

Укажите **не менее одного** варианта ответа.

— `int a [3]`

+ `int a [ 'f' ]`

— `int a []`

— `int a [3] = {1}`

21. Пусть `int a [5] = {10, 11, 12, 13, 14}`. Чему равно значение элемента `a [3]`?

Укажите **не менее одного** варианта ответа.

— 3

— 12

+ 13

— 14

Если в программе на языке C ++ в производном классе переопределена операция `new to ...`

Укажите **не менее одного** варианта ответа.

все объекты этого класса и все объекты классов, выведенных из него, будут использовать эту операцию независимо от зоны видимости, в которой она переопределена.
производные от этого класса могут использовать глобальную операцию применив операцию базовый_класс :: new. +
операцию new нельзя переопределить.
в любом случае эта операция будет доступна только в пределах класса-потомка.

23. Какой из перечисленных методов может быть конструктором для класса String в языке C ++?

Укажите **не менее одного** варианта ответа.

String * String ()
void String ()
String (String & s) +
const String (int a)

24. Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам?

Укажите **не менее одного** варианта ответа.

Шаблонная.
Полиморфная.
Дружеская. +
Статическая.

25. Вызовет данный код ошибку компиляции? class Rectangle public: int a, b; int sum (); int square (); ~ Rect (); };

Укажите **не менее одного** варианта ответа.

Ошибки нет, все записано верно.
Ошибки являются: имя деструктора должно совпадать с именем класса. +
Ошибки являются: имя деструктора не может начинаться с маленькой буквы.
Ошибки являются: никакой идентификатор в C ++ не может начинаться со знака «~».

26. Принцип объектно-ориентированного программирования, заключается в объединении атрибутов и методов объекта с целью обеспечения сохранности данных, называется:

Укажите **не менее одного** варианта ответа.

Наследование
Сочетание
Инициализация
Инкапсуляция +

27. В программе описано абстрактный класс A и производный от этого класса класс A1. Какой из записей заведомо неверный?

Укажите **не менее одного** варианта ответа.

A * a = new A +
A1 a1
A1 a1; A & a = a1
A1 a1; A1 a2

28. Что такое заголовочные файлы?

Укажите **не менее двух** вариантов ответа.

— Название программы, указывается при сохранении
— Название главной функции или функции пользователя
— Модули, сохраняют заголовки функций +
— Стандартные библиотеки, расположенные в папке include +

29. Укажите правильное объявление виртуального метода, который принимает одно целочисленное значение и возвращает void.

Укажите **не менее одного** варианта ответа.

— virtual void SomeFunction (int x) +
— void SomeFunction (int x) virtual
— virtual SomeFunction (int x)
— virtual void SomeFunction (int * x)

30. Какой из вариантов записи абстрактного класса в C ++ является правильным?

Укажите **не менее одного** варианта ответа.

abstract class A {virtual int f () = 0;}

class A {virtual int f () = 0;} +

class A {virtual int f () = 0;} abstract

class A {virtual int f ();}

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ

ответов на тестовые вопросы тестирования по итогам освоения дисциплины

- оценка «отлично» выставляется обучающемуся, если получено более 81% правильных ответов.
- оценка «хорошо» - получено от 71 до 80% правильных ответов.
- оценка «удовлетворительно» - получено от 61 до 70% правильных ответов.
- оценка «неудовлетворительно» - получено менее 61% правильных ответов.

**Вопросы и практические задания
для проведения промежуточной аттестации по дисциплине «Теоретические основы
программирования»
(экзамен)**

Вопросы

1. Что такое программирование?
2. Как работает программный код? Что такое отладка?
3. Назовите типы ошибок, которые могут возникнуть в программе. Объясните одну из них.
4. Что означает «Поддерживать и обновлять программу»?
5. Объясните понятие «переменные»? Зарезервированные слова в программировании?
6. Что такое цикл? Назовите разные виды циклов
7. Чем отличаются for и while? Что такое вложенный цикл?
8. Когда применяют арифметические и реляционные операторы? Для чего нужны логические операторы?
9. Низкоуровневый и высокоуровневый языки программирования: отличие
10. Общее понятие типов данных. Что определяет тип данных?
11. Обозначьте основные положения структурного программирования.
12. Как обозначаются управляющие структуры и инструкции языка C++?
13. Что такое инкапсуляция и наследование в C++?
14. Что такое пространство имен?
15. Каковы наиболее важные различия между C и C++?
16. Что такое деструктор?
17. Какие различия отделяют структуру от класса в C++?
18. Функция вычисляет произведение двух чисел. Исходные данные вводятся с клавиатуры. Какие проверки целесообразно ввести в программе?
19. Что из себя представляет динамическое выделение памяти?
20. В чем разница между указателями и ссылками в C++?

Практические задания

1. Каков результат следующей программы?

```
#include <iostream>
using namespace std;
int main()
{
    int numbers[5], sum = 0;
    cout << "Enter 5 numbers: ";
    for (int i = 0; i < 5; ++i)
    {
        cin >> numbers[i];
        sum += numbers[i];
    }
    cout << "Sum = " << sum << endl;
    return 0;
}
```

2. В программе для определения является ли год високосным, подключите пространство имен. Преобразуйте программу.

```
#include <iostream>
int main() {
    int year;
    std::cin >> year;

    if (year % 400 == 0) {
        std::cout << "YES\n";
    } else if (year % 100 == 0) {
        std::cout << "NO\n";
    } else if (year % 4 == 0) {
        std::cout << "YES\n";
    } else {
        std::cout << "NO\n";
    }
}
```

```
}
```

3. В представленном коде дублируются ветви условий. Упростите его.

```
#include <iostream>
int main() {
    int year;
    std::cin >> year;

    if (year % 400 == 0) {
        std::cout << "YES\n";
    } else if (year % 100 == 0) {
        std::cout << "NO\n";
    } else if (year % 4 == 0) {
        std::cout << "YES\n";
    } else {
        std::cout << "NO\n";
    }
}
```

4. Задача «Сумма чисел». Условие: Вам даны два целых числа. Напечатайте их сумму.

Формат ввода: Вводятся два числа, по модулю не превосходящие миллиарда.

Формат вывода: Напечатайте сумму этих чисел. В конце поставьте перевод строки.

5. Составьте программу, выводящую на экран квадраты чисел от 10 до 20 включительно.

6. Даны натуральные числа от 35 до 87. Вывести на консоль те из них, которые при делении на 7 дают остаток 1, 2 или 5.

7. Поясните, что выполняет данный программный код

```
int p=1;
for (int i=1; i<n; i+=2)
{
    p*=a[i];
}
cout<<"answer: "<<p<<endl;
```

8. Поясните, что выполняет данный программный код

```
int max=0;
for (int i=1; i<n; i++) { if (a[i]>max)
    max=a[i];
}
cout<<"max: "<<max<<endl;
```

9. Поясните, что выполняет данный программный код

```
int imin=-1;
for (int i=0; i<n; i++) {
    if (!(a[i]%2) && (imin==-1 || a[imin]>a[i]))
        imin=i;
}
if (imin==-1)
    cout<<a[0];
else
    cout<<a[imin]<<endl;
```

10. Поясните, что выполняет данный программный код

```
int imax=0, imin=0;
for (int i=1; i < n; i++) {
    if (a[i]>a[imax])
        imax=i;
    if (a[i]<a[imin])
        imin=i;
}
```

cout<<"answer: "<<imin+imax<<endl;

ПЛАНОВАЯ ПРОЦЕДУРА проведения экзамена

Нормативная база проведения промежуточной аттестации обучающихся по результатам изучения дисциплины:	
1) действующее «Положение о текущем контроле успеваемости, промежуточной аттестации обучающихся по программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры и среднего профессионального образования в ФГБОУ ВО Омский ГАУ»	
Основные характеристики промежуточной аттестации обучающихся по итогам изучения дисциплины	
Цель промежуточной аттестации -	установление уровня достижения каждым обучающимся целей обучения по данной дисциплине
Форма промежуточной аттестации -	Экзамен
Место экзамена в графике учебного процесса:	1) подготовка к экзамену и сдача экзамена осуществляется за счёт учебного времени (трудоемкости), отведённого на экзаменационную сессию для обучающихся, сроки которой устанавливаются приказом по университету
	2) дата проведения экзамена определяется графиком сдачи экзаменов, утверждаемым приказом ректора
Форма экзамена -	<i>Письменный</i>
Время проведения экзамена	Время и место проведения экзамена определяется графиком сдачи экзаменов, утверждаемым деканом факультета

ШКАЛА И КРИТЕРИИ ОЦЕНКИ

Результаты экзамена определяют оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» и объявляют в день экзамена.

Оценку «отлично» выставляют обучающемуся, глубоко и прочно освоившему теоретический и практический материал дисциплины. Ответ должен быть логичным, грамотным. Обучающемуся необходимо показать знание не только основного, но и дополнительного материала, быстро ориентироваться, отвечая на дополнительные вопросы. Обучающийся должен свободно справляться с поставленными задачами, правильно обосновывать принятые решения.

Оценку «хорошо» заслуживает обучающийся, твердо знающий программный материал дисциплины, грамотно и по существу излагающий его. Не следует допускать существенных неточностей при ответах на вопросы, необходимо правильно применять теоретические положения при решении практических задач, владеть определенными навыками и приемами их выполнения.

Оценку «удовлетворительно» получает обучающийся, который имеет знания только основного материала, но не усвоил его детали, испытывает затруднения при решении практических задач. В ответах на поставленные вопросы обучающимся допущены неточности, даны недостаточно правильные формулировки, нарушена последовательность в изложении программного материала.

Оценка «неудовлетворительно» говорит о том, что обучающийся не знает значительной части материала по дисциплине, допускает существенные ошибки в ответах, не может решить практические задачи или решает их с затруднениями.

ЛИСТ РАССМОТРЕНИЙ И ОДОБРЕНИЙ

Фонд оценочных средств учебной дисциплины Б1.О.11 Теоретические основы
программирования
в составе ОПОП 09.04.02 Информационные системы и технологии

1. Рассмотрен и одобрен в качестве базового варианта:

а) На заседании обеспечивающей кафедры экономики, бухгалтерского учета и финансового контроля
протокол № 11 от 19.05.2022.

Зав. кафедрой, канд. экон. наук, доцент

 О.А. Блинов

б) На заседании методической комиссии по направлению 09.04.02 Информационные системы и
технологии

протокол № 9 от 24.05.2022.

Председатель МКН – 09.04.02, канд. экон. наук

 С.А. Нардина

2. Рассмотрен и одобрен внешним экспертом

Директор ООО «Епортал»



И.И. Линник

М.П.

ИЗМЕНЕНИЯ И ДОПОЛНЕНИЯ
к фонду оценочных средств учебной дисциплины
Б1.О.11 Теоретические основы программирования
в составе ОПОП 09.04.02 Информационные системы и технологии

Ведомость изменений

Срок, с которого вводится изменение	Номер и основное содержание изменения и/или дополнения	Отметка об утверждении/ согласовании изменений	
		инициатор изменения	руководитель ОПОП/ председатель МК/ПЦМК