Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал
федерального государственного автономного образовательного учреждения
высшего образования
**«Пермский национальный исследовательский
политехнический университет»**

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
для проведения промежуточной аттестации обучающихся по дисциплине
«Основы алгоритмизации и программирования»
Приложение к рабочей программе дисциплины

Направление подготовки:	09.03.01 Информатика и вычислительная техника
Направленность (профиль) образовательной программы:	Компьютерные системы
Квалификация выпускника:	«Бакалавр»
Выпускающая кафедра:	Общенаучных дисциплин
Форма обучения:	Очная, очно-заочная, заочная

Курс: 1 **Семестр**: 1,2

Трудоёмкость:
Кредитов по рабочему учебному плану: 9 ЗЕ
Часов по рабочему учебному плану: 324 ч.

Форма промежуточной аттестации:
Экзамен: 1,2 семестр

Лысьва 2024

Фонд оценочных средств для проведения промежуточной аттестации обучающихся для проведения промежуточной аттестации обучающихся по дисциплине является частью (приложением) к рабочей программе дисциплины. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине разработан в соответствии с общей частью фонда оценочных средств для проведения промежуточной аттестации основной образовательной программы, которая устанавливает систему оценивания результатов промежуточной аттестации и критерии выставления оценок**.** Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине устанавливает формы и процедуры текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине.

1. Перечень контролируемых результатов обучения по дисциплине, объекты оценивания и виды контроля

Согласно РПД освоение учебного материала дисциплины запланировано в течение двух семестров (1-го и 2 семестра учебного плана всех форм обучения). В каждом разделе предусмотрены аудиторные лекционные и лабораторные занятия, а также самостоятельная работа студентов. В рамках освоения учебного материала дисциплины формируется компоненты компетенций *знать, уметь, владеть*, указанные в РПД, которые выступают в качестве контролируемых результатов обучения по дисциплине (табл. 1.1).

Контроль уровня усвоенных знаний, освоенных умений и приобретенных владений осуществляется в рамках текущего, рубежного и промежуточного контроля при изучении теоретического материала, сдаче отчетов по лабораторным работам и экзамена. Виды контроля сведены в таблицу 1.1.

Таблица 1.1. Перечень контролируемых результатов обучения по дисциплине

Контролируемые результаты обучения по дисциплине (ЗУВы)	**Вид контроля**					
	Текущий		**Рубежный**		**Итоговый**	
	С	**ТО**	**ОЛР**	**Т/КР**		**Экзамен**
Усвоенные знания						
З.1 знать основы математики, физики, вычислительной техники и программирования, необходимые для решения прикладных задач при проектировании и реализации алгоритмов		ТО	ОЛР	Т		ТВ
З.2 знать алгоритмические языки программирования семейства С, современные интегрированные среды разработки программного обеспечения, операционные системы семейства Windows и Linux		ТО	ОЛР	Т		ТВ
Освоенные умения						
У.1 уметь решать задачи по проектированию и реализации алгоритмов с применением естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования.			ОЛР	Т		ПЗ
У.2 уметь составлять алгоритмы без привязки к отдельным языкам программирования, писать и отлаживать код на языке С++, тестировать работоспособность программы при помощи средств разработки, интегрировать программные модули.			ОЛР	Т		ПЗ

Контролируемые результаты обучения по дисциплине (ЗУВы)	Вид контроля					
	Текущий		Рубежный		Итоговый	
	С	ТО	ОЛР	Т/КР		Экзамен
Приобретенные владения						
В.1 владеть навыками теоретического и экспериментального исследования алгоритмов			ОЛР			КЗ
В.2 владеть навыками отладки и тестирования работоспособности программы; языком программирования C#			ОЛР			КЗ

ТО – коллоквиум (теоретический опрос); ОЛР – отчет по лабораторной работе; Т/КР – рубежное тестирование (контрольная работа); ТВ – теоретический вопрос; ПЗ – практическое задание; КЗ – комплексное задание экзамена.

Итоговой оценкой достижения результатов обучения по дисциплине является промежуточная аттестация в форме экзамена, проводимая с учетом результатов текущего и рубежного контроля.

1. Виды контроля, типовые контрольные задания и шкалы оценивания результатов обучения

Текущий контроль успеваемости имеет целью обеспечение максимальной эффективности учебного процесса, управление процессом формирования заданных компетенций обучаемых, повышение мотивации к учебе и предусматривает оценивание хода освоения дисциплины. В соответствии с Положением о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по образовательным программам высшего образования – программам бакалавриата, специалитета и магистратуры в ПНИПУ предусмотрены следующие виды и периодичность текущего контроля успеваемости обучающихся:

- входной контроль, проверка исходного уровня подготовленности обучаемого и его соответствия предъявляемым требованиям для изучения данной дисциплины;

- текущий контроль усвоения материала (уровня освоения компонента «знать» заданных компетенций) на каждом групповом занятии и контроль посещаемости лекционных занятий;

- промежуточный и рубежный контроль освоения обучаемыми отдельных компонентов «знать», «уметь» заданных компетенций путем компьютерного или бланочного тестирования, контрольных опросов, контрольных работ (индивидуальных домашних заданий), защиты отчетов по лабораторным работам, рефератов, эссе и т.д.

Рубежный контроль по дисциплине проводится на следующей неделе после прохождения модуля дисциплины, а промежуточный – во время каждого контрольного мероприятия внутри модулей дисциплины;

- межсессионная аттестация, единовременное подведение итогов текущей успеваемости не менее одного раза в семестр по всем дисциплинам для каждого направления подготовки (специальности), курса, группы;

- контроль остаточных знаний.

2.1. Текущий контроль усвоения материала

Текущий контроль усвоения материала в форме выборочного теоретического опроса студентов проводится по каждой теме. Результаты по 4-

балльной шкале оценивания заносятся в книжку преподавателя и учитываются в виде интегральной оценки при проведении промежуточной аттестации.

2.2. Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, освоенных умений и приобретенных владений (табл. 1.1) проводится в форме защиты лабораторных работ и тестирования (после изучения каждого раздела учебной дисциплины).

2.2.1. Защита лабораторных работ

Всего запланировано 13 лабораторных работ. Типовые темы лабораторных работ приведены в РПД.

Защита лабораторной работы проводится индивидуально каждым студентом или группой студентов. Типовые шкала и критерии оценки приведены в общей части ФОС образовательной программы.

2.2.2. Рубежное тестирование

Согласно РПД запланировано рубежное тестирование после освоения студентами учебных разделов дисциплины.

Типовые шкала и критерии оценки рубежного тестирования приведены в общей части ФОС образовательной программы.

Типовые тестовые задания (Т)

Вариант 1

1) Последовательность шагов, предназначенная для решения конкретной задачи – это …
2) Именованное хранилище для данных, которые могут изменяться в процессе выполнения программы – это …
3) Простой тип данных, предназначенный для хранения знаковых целых чисел – это …
4) Конструкция, позволяющая повторять определённый блок кода до выполнения определённого условия – это …
5) Какие виды циклов представлены в C#?
6) Конструкция, позволяющая выполнить определённые действия в зависимости от выполнения определённых условий – это …
7) Участок кода, который может быть вызван из другого места программы для выполнения определённой задачи – это …
8) Текстовые объяснения в коде, которые не влияют на его выполнение, но помогают разработчикам понять его логику и цели – это …
9) Какие два основных вида оценки сложности алгоритмов существуют?
10) Именованная структура для хранения заранее известного количества однотипных элементов – это …

Ответы:

1) Алгоритм
2) Переменная
3) Integer
4) Цикл

5) For, while, do while, foreach
6) Условный оператор
7) Функция
8) Комментарий
9) Объемная и временная
10) Массив

Вариант 1

1) Тип данных (в C#), предназначенный для хранения целых чисел и занимающий в памяти 4 байта – это …
2) Свойство алгоритма, характеризующее его применимость к различным наборам исходных данных, называется …
3) Постоянные значения, которые известны во время компиляции и не изменяются во время выполнения программы – это …
4) Конструкция (структура), позволяющая выбрать один из двух (в общем случае) блоков кода, в зависимости от выполнения определённого условия – это …
5) Оператор множественного выбора на языке C# выглядит как …
6) Оператор, позволяющий выполнить блок кода заранее известное количество раз – это …
7) Участок кода, который может быть вызван из другого места программы для выполнения определённой задачи по его имени – это …
8) Точка входа в приложение C# – это метод (функция) …
9) Оценка сложности алгоритма, основанная на измерении количества памяти, необходимого для его реализации, называется …?
10) Именованная структура для хранения заранее известного количества однотипных элементов – это …

Ответы:

1) int (integer)
2) Массовость (Универсальность)
3) Константа (-ы)
4) Ветвление (Ветвящаяся структура)
5) switch (switch case)
6) for (цикл)
7) Функция (метод)
8) main
9) Объемная
10) Массив

Типовые шкала и критерии оценки результатов рубежной контрольной работы приведены в общей части ФОС образовательной программы.

2.3. Промежуточная аттестация (итоговый контроль)

Допуск к промежуточной аттестации осуществляется по результатам текущего и рубежного контроля. Условиями допуска являются успешная сдача всех

лабораторных работ и положительная интегральная оценка по результатам текущего и рубежного контроля.

Промежуточная аттестация, согласно РПД, проводится в форме экзамена по дисциплине устно по билетам. Билет содержит теоретические вопросы (ТВ) для проверки усвоенных знаний, практические задания (ПЗ) для проверки освоенных умений и комплексные задания (КЗ) для контроля уровня приобретенных владений всех заявленных компетенций.

Билет формируется таким образом, чтобы в него попали вопросы и практические задания, контролирующие уровень сформированности ***всех*** заявленных компетенций. Форма билета представлена в общей части ФОС образовательной программы.

2.3.1. Типовые вопросы и задания для экзамена по дисциплине

Типовые вопросы для контроля усвоенных знаний:

1. Классификация языков программирования.
2. Понятие алгоритма. Свойства и виды алгоритмов. Способы описания алгоритмов. Основные конструкции алгоритмического языка: линейный алгоритм, ветвление, цикл
3. Этапы решения задач с помощью ЭВМ: постановка задачи, создание модели, алгоритм, кодирование, анализ результатов
4. Виды, классификации и назначение языков программирования. Стандарты языков программирования. Среда программирования. Компиляторы и интерпретаторы
5. Жизненный цикл программы. Программный продукт и его характеристики
6. Переменные и константы. Объявление объектов данных. Внутреннее представление данных в памяти компьютера
7. Типы данных. Виды типов данных
8. Понятие системы программирования. Основные функции системы программирования
9. Типы данных, определяемые программистом. Перечисляемый и интервальный типы
10. Форматированный вывод данных
11. Приоритеты выполнения действий в выражениях
12. Составление и отладка программ с использованием арифметических выражений.
13. Общие сведения о подпрограммах. Процедуры и функции. Определение и вызов подпрограмм. Описание, выполнение процедур и функций. Виды параметров в подпрограммах. Области видимости переменных. Механизм передачи параметров.
14. Основы структурного программирования. Методы структурного программирования
15. Объявление массива. Инициализация. Действия над массивами. Заполнение массива данными. Ввод и вывод одномерных и двумерных массивов. Стандартные функции для массива целых и вещественных чисел. Обработка массива. Удаление и вставка элементов в массив

16. Символьный и строковый типы. Объявление типов. Поиск, удаление, замена и добавление символов в строке. Операции со строками

17. Стандартные функции и процедуры для работы со строками. Массив символов, строки и их обработка

18. Типы файлов. Организация доступа к файлам. Файлы последовательного доступа: открытие и закрытие, запись в файл и чтение из файла. Файлы произвольного доступа.

19. Работа с массивами. Способы ввода элементов в массив – ввод с клавиатуры, через формулы, оператор присваивания или случайным образом Randomize.

20. Примеры использования строковых типов данных. Отладка и тестирование программ с использованием строковых типов данных.

21. Выполнение анализа процедур при работе со множествами. Представление множеств линейными массивами.

22. Событийно-управляемая модель программирования. Компонентно-ориентированный подход. Классы объектов. Компоненты и их свойства

23. Формы и компоненты. Принципы визуального программирования. Свойства компонентов. Создание простого приложения

24. Переменные. Типы данных в C#.

25. Структура программы. Команда присваивания в C#.

26. Ввод-вывод данных. Формат выводимых данных.

27. Алгоритм линейной структуры в C#.

28. Структура IF, классификация в C#. Примеры.

29. Структура switch(выбор) и ее программирование в C#. Примеры.

30. Алгоритмы циклической итерационной структуры. Оператор цикла While в C#. Примеры использования.

31. Алгоритмы циклической итерационной структуры. Оператор цикла do… while в C#. Примеры использования.

32. Алгоритмы циклической итерационной структуры. Оператор цикла For в C#. Примеры использования.

33. Операторы break и continue в C#. Примеры использования.

34. Одномерные массивы в C#. Задание массивам первоначальных значений.

35. Операции над массивами. Ввод-вывод массивов в C#.

36. Случайные числа в языке программирования C#.

37. Понятие подпрограммы в C#. Описание подпрограммы.

38. Формальные и фактические параметры в C#.

39. Понятие о локальных и глобальных переменных в C#.

40. Основные математические функции в C#. Примеры.

41. Определение алгоритма. Свойства алгоритма. Формы записи алгоритмов. Примеры.

42. Запись алгоритмов блок-схемами. Основные элементы блок-схем

43. Алгоритмы с ветвлением. Пример алгоритма.

44. Алгоритм цикла с предусловием. Пример алгоритма. Алгоритм цикла с постусловием. Пример алгоритма.

45. Алгоритм цикла с управляющей переменной. Пример алгоритма.

Типовые вопросы и практические задания для контроля освоенных умений:

1. В массив $A[N]$ занесены натуральные числа. Найти сумму тех элементов, которые кратны данному K.

2. В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.

3. Дана последовательность целых чисел a_1, a_2, …, a_n. Выяснить, какое число встречается раньше — положительное или отрицательное.

4. Дана последовательность действительных чисел a_1, a_2, …, a_n. Выяснить, будет ли она возрастающей.

5. Дана последовательность натуральных чисел a_1, a_2, …, a_n. Создать массив из четных чисел этой последовательности. Если таких чисел нет, то вывести сообщение об этом факте.

6. В одномерном массиве все отрицательные элементы переместить в начало массива, а остальные — в конец с сохранением порядка следования. Дополнительный массив заводить не разрешается.

7. Составить программу, которая заполняет квадратную матрицу порядка n натуральными числами 1, 2, 3, …, n^2, записывая их в нее «по спирали».

Например, для $n = 5$ получаем следующую матрицу:

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

8. Дана действительная квадратная матрица порядка $2n$. Получить новую матрицу, переставляя ее блоки размера $n \times n$ по часовой стрелке, начиная с блока в левом верхнем углу.

9. Составить программу, позволяющую в одномерном массиве вычислить сумму модулей отрицательных элементов массива

10. Составить программу, позволяющую в одномерном массиве вычислить количество элементов массива, не принадлежащих интервалу (a, b)

11. В данном линейном массиве четные положительные элементы заменить на 1, а нечетные отрицательные — на -1. Остальные оставить без изменения

12. В одномерном массиве с четным количеством элементов ($2N$) находятся координаты N точек плоскости. Они располагаются в следующем порядке: x_1, y_1, x_2, y_2, x_3, y_3, и т.д.

а) Определить минимальный радиус окружности с центром в начале координат, которая содержит все точки.

б) Определить номера точек, которые могут являться вершинами квадрата.

в) Определить номера точек, которые могут являться вершинами равнобедренного треугольника.

г) Найти номера самых удаленных друг от друга точек и наименее удаленных друг от друга точек.

13. В массив A[N] занесены натуральные числа. Найти сумму тех элементов, которые кратны данному K.

14. В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.

15. Дана последовательность целых чисел a_1, a_2, …, a_n. Выяснить, какое число встречается раньше — положительное или отрицательное.

16. Дана последовательность действительных чисел a_1, a_2, …, a_n. Выяснить, будет ли она возрастающей.

17. Дана последовательность натуральных чисел a_1, a_2, …, a_n. Создать массив из четных чисел этой последовательности. Если таких чисел нет, то вывести сообщение об этом факте.

18. В одномерном массиве все отрицательные элементы переместить в начало массива, а остальные — в конец с сохранением порядка следования. Дополнительный массив заводить не разрешается.

19. Составить программу, позволяющую в одномерном массиве вычислить сумму модулей отрицательных элементов массива

20. Составить программу, позволяющую в одномерном массиве вычислить количество элементов массива, не принадлежащих интервалу (a, b)

21. В данном линейном массиве четные положительные элементы заменить на 1, а нечетные отрицательные — на -1. Остальные оставить без изменения

22. В массив $A[N]$ занесены натуральные числа. Найти сумму тех элементов, которые кратны данному K.

23. В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.

24. Дана последовательность целых чисел a_1, a_2, …, a_n. Выяснить, какое число встречается раньше — положительное или отрицательное.

25. Дана последовательность действительных чисел a_1, a_2, …, a_n. Выяснить, будет ли она возрастающей.

26. Дана последовательность натуральных чисел a_1, a_2, …, a_n. Создать массив из четных чисел этой последовательности. Если таких чисел нет, то вывести сообщение об этом факте.

27. В одномерном массиве все отрицательные элементы переместить в начало массива, а остальные — в конец с сохранением порядка следования. Дополнительный массив заводить не разрешается.

28. Составить программу, позволяющую в одномерном массиве вычислить сумму модулей отрицательных элементов массива

29. Составить программу, позволяющую в одномерном массиве вычислить количество элементов массива, не принадлежащих интервалу (a, b)

30. В данном линейном массиве четные положительные элементы заменить на 1, а нечетные отрицательные — на -1. Остальные оставить без изменения

Типовые комплексные задания для контроля приобретенных владений:

Решить задачу

Вариант 1

1) Создать базовый класс с тремя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с тремя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Масса». Наследует базовый класс. Поля определяют центнеры, килограммы, граммы. Два конструктора: без параметров – переопределяется конструктор базового класса (на данные накладываются ограничения, но они по-прежнему формируются случайно); с тремя параметрами – обращается к соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода «ЦЦ ц КК кг ГГГ г»; б) сложение (сложить две массы – получить третью массу); в) вычитание (вычесть ккг, где k – любое неотрицательное целое число, в т.ч. больше 100).

3) Класс «Точка в трехмерном пространстве». Наследует базовый класс. Поля определяют координаты точки. Два конструктора: без параметров и с тремя параметрами – обращается к соответствующим конструкторам базового класса. Методы: а) сдвиг на вектор (к координатам точки прибавляются координаты вектора); б) расстояние между двумя точками; в) точка, симметричная данной относительно начала координат. Вывод используется из базового класса.

Вариант 2

1) Создать базовый класс с двумя символьными полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с двумя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Строка». Наследует базовый класс. Поле — двухсимвольная строка. Два конструктора: без параметров – использует конструктор базового класса (символы сцепляются); с двумя параметрами – использует конструктор базового класса (символы сцепляются). Методы: а) вывод (переопределить метод базового класса); б) сравнение; в) вычитание (уменьшить код каждого символа на k. Если получается отрицательный код – ответ – пустая строка).

3) Класс «Массив, элементами которого являются пары символов». Наследует базовый класс. Два конструктора: заполнение случайно и копированием (задействовать конструкторы базового класса). Методы: а) сортировка по первому символу, но пара переставляется целиком; б) поиск пары с максимальной и минимальной суммой символов (суммируются коды); в) изменение регистра (если символ – буква), - на противоположный); г) переопределить метод вывода (задействовать при этом метод вывода из базового класса).

Вариант 3

1) Создать базовый класс с двумя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с двумя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Обыкновенная дробь». Наследует базовый класс. Поля определяют числитель и знаменатель. Два конструктора: без параметров – переопределяется конструктор базового класса (на данные накладываются ограничения, но они по-прежнему формируются случайно); с двумя параметрами – обращается к

соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода «Ч / З»; б) сложение; в) вычитание.

3) Класс «Прямоугольник». Наследует базовый класс. Поля определяют ширину и длину прямоугольника. Два конструктора: без параметров и с двумя параметрами – обращается к соответствующим конструкторам базового класса. Методы: а) периметр; б) площадь; в) выбор из двух прямоугольников прямоугольника с наибольшей площадью. Вывод используется из базового класса.

Вариант 4

1) Создать базовый класс с тремя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с тремя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Окружность на плоскости». Наследует базовый класс. Поля определяют координаты центра окружности и радиус. Два конструктора: без параметров, с тремя параметрами – обращаются к соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода «(x; y), r»; б) площадь; в) длина окружности; г) окружность минимального целочисленного радиуса такая, что центр ее располагается в центре первой заданной окружности, и она включает вторую заданную окружность.

3) Класс «Точка в трехмерном пространстве». Наследует базовый класс. Поля определяют координаты точки. Два конструктора: без параметров и с тремя параметрами – обращается к соответствующим конструкторам базового класса. Методы: а) точка с целочисленными координатами, которая располагается примерно посередине между данной и началом координат; б) точки, симметричные данной относительно каждой из координатных осей; в) точки, являющиеся проекцией данной на каждую из координатных осей. Вывод используется из базового класса.

Вариант 5

1) Создать базовый класс с двумя вещественными полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с двумя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Треугольник на плоскости». Наследует базовый класс. Добавляются ещё 4 вещественных поля. Пары чисел обозначают координаты точки на плоскости. Конструкторы тоже должны быть ориентированы на работу с ещё 4 дополнительными полями. Переопределить метод вывода. Методы: а) проверка (логический метод), являются ли заданные точки вершинами треугольника (т.е. не лежат ли на одной прямой); б) периметр треугольника; в) площадь треугольника.

3) Класс «Массив треугольников». Наследует класс «Треугольник на плоскости». Метод: а) треугольник наибольшей площади; б) треугольник наименьшего периметра.

Вариант 6

1) Создать базовый класс с четырьмя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными;

второй – с четырьмя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Отрезок на плоскости». Наследует базовый класс. Первые два числа обозначают координаты одного, следующие два — другого конца отрезка. Переопределить вывод. Методы: а) длина отрезка; б) отрезок, симметричный данному относительно оси Ox; в)отрезок, симметричный данному относительно начала координат.

3) Класс «Прямоугольник на плоскости со сторонами, параллельными координатным осям». Наследует базовый класс. Первые два числа обозначают координаты левого верхнего, следующие два — координаты правого нижнего углов прямоугольника. Конструктор без параметров переопределить, чтобы координаты генерировались корректно. Переопределить вывод. Методы: а) периметр; б) площадь; в) выбор из двух прямоугольников прямоугольника с наибольшей площадью.

Вариант 7

1) Создать базовый класс с тремя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с тремя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Обыкновенная дробь с целой частью». Наследует базовый класс. Поля определяют целую часть, числитель и знаменатель. Два конструктора: без параметров – переопределяется конструктор базового класса (на данные накладываются ограничения, но они по-прежнему формируются случайно); с тремя параметрами – обращается к соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода «Ц Ч / З»; б) сокращение дроби; в) сложение; г) вычитание.

3) Класс «Треугольник на плоскости». Наследует базовый класс. Поля обозначают длины сторон треугольника. Методы: а) проверка (логический метод), являются ли заданные числа длинами сторон треугольника (выполняется ли неравенство треугольника); б) периметр треугольника; в) площадь треугольника.

Вариант 8

1) Создать базовый класс с тремя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с тремя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Время». Наследует базовый класс. Поля определяют часы, минуты, секунды. Два конструктора: без параметров – переопределяется конструктор базового класса (на данные накладываются ограничения, но они по-прежнему формируются случайно); с тремя параметрами – обращается к соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода ЧЧ:ММ:СС; б) сложение (сложить два времени – получить третье время); в) вычитание (вычесть kсекунд, где k – любое неотрицательное целое число, в т.ч. больше 60).

3) Класс «Вектор в трехмерном пространстве». Наследует базовый класс. Поля определяют координаты вектора. Два конструктора: без параметров и с тремя

параметрами – обращается к соответствующим конструкторам базового класса. Методы: а) сложение векторов; б) вычитание векторов; в) умножение вектора на целое число. Вывод используется из базового класса.

Вариант 9

1) Создать базовый класс с двумя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с двумя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Масса». Наследует базовый класс. Поля определяют тонны, килограммы. Два конструктора: без параметров – переопределяется конструктор базового класса (на данные накладываются ограничения, но они по-прежнему формируются случайно); с двумя параметрами – обращается к соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода «ТТТТ т ККК кг»; б) сложение (сложить две массы – получить третью массу); в) вычитание (вычесть ккг, где k – любое неотрицательное целое число, в т.ч. больше 1000).

3) Класс «Точка в двумерном пространстве». Наследует базовый класс. Поля определяют координаты точки. Два конструктора: без параметров и с двумя параметрами – обращается к соответствующим конструкторам базового класса. Методы: а) сдвиг на вектор (к координатам точки прибавляются координаты вектора); б) расстояние между двумя точками; в) точка, симметричная данной относительно начала координат. Вывод используется из базового класса.

Вариант 10

1) Создать базовый класс с тремя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с тремя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс «Квадрат на плоскости со сторонами, параллельными осям координат».Наследует базовый класс. Поля определяют: первые два числа — координаты левого верхнего угла, третье — длина стороны. Переопределить метод вывода. Методы: а) периметр; б) площадь; в) квадрат, симметричный данному относительно начала координат.

3) Класс «Время». Наследует базовый класс. Поля определяют часы, минуты, секунды. Два конструктора: без параметров – переопределяется конструктор базового класса (на данные накладываются ограничения, но они по-прежнему формируются случайно); с тремя параметрами – обращается к соответствующему конструктору базового класса. Методы: а) вывод (переопределить метод базового класса) – формат вывода ЧЧ:ММ:СС; б) сравнение (результат — «раньше», «позже», «равны»); в) умножение— умножить нанатуральное число k(например, для 12:02:22 при умножении на 2 получим 00:04:44).Указание. Действие выполняем над временем в секундах и затем переводим обратно в часы, минуты, секунды, убрав целые сутки, если они прошли.

Вариант 11

1) Создать базовый класс с одним целым полем. Разработать два конструктора: первый – без параметров – заполняет поле случайным положительным числом; второй – с параметром – конструктор копирования. Разработать метод: вывод поля.

2) Класс «Натуральное число». Наследует базовый класс. Поле определяет натуральное число. Методы: а) количество цифр; б) сумма цифр; в) является ли палиндромом (true, false); г) «перевертыш» числа.

3) Класс «Массив натуральных чисел». Наследует класс «Натуральное число». Переопределить метод вывода. Отсортировать по количеству цифр и сумме цифр (т.е. в каждой группе с одинаковым количеством цифр сортируем по сумме цифр).

Вариант 12

1) Создать базовый класс с двумя целыми полями. Разработать два конструктора: первый – без параметров – заполняет поля случайными данными; второй – с двумя параметрами – конструктор копирования. Разработать метод: вывода полей.

2) Класс«Прямоугольный треугольник». Наследует базовый класс. Поля определяют длины катетов. Два конструктора: без параметров – переопределяется конструктор базового класса; с двумя параметрами – обращается к соответствующему конструктору базового класса. Методы: а) гипотенуза; б) периметр; в) площадь. Примечание. Катеты должны быть таковы, чтобы гипотенуза тоже была целым числом!

3) Класс «Длина». Наследует базовый класс. Поля обозначают длину в метрах и сантиметрах. Методы: а) вывод (переопределить метод базового класса) – формат вывода «ММММмСССм»; б) сложение (сложить две длины – получить третью длину); в) вычитание (вычесть kсм, где k – любое неотрицательное целое число, в т.ч. больше 99).

Перечень типовых ситуационных заданий и кейсов для проверки умений и владений представлен в приложении 1. *Полный перечень теоретических вопросов и практических заданий в форме утвержденного комплекта экзаменационных билетов хранится на выпускающей кафедре.*

2.3.2. Шкалы оценивания результатов обучения на экзамене

Оценка результатов обучения по дисциплине в форме уровня сформированности компонентов *знать, уметь, владеть* заявленных компетенций проводится по 4-х балльной шкале оценивания путем выборочного контроля во время экзамена.

Типовые шкала и критерии оценки результатов обучения при сдаче экзамена для компонентов *знать, уметь и владеть* приведены в общей части ФОС образовательной программы.

3. Критерии оценивания уровня сформированности компонентов и компетенций

3.1. Оценка уровня сформированности компонентов компетенций

При оценке уровня сформированности компетенций в рамках выборочного

контроля при экзамене считается, что *полученная оценка за компонент проверяемой в билете компетенции обобщается на соответствующий компонент всех компетенций, формируемых в рамках данной учебной дисциплины*.

Типовые критерии и шкалы оценивания уровня сформированности компонентов компетенций приведены в общей части ФОС образовательной программы.

3.2. Оценка уровня сформированности компетенций

Общая оценка уровня сформированности всех компетенций проводится путем агрегирования оценок, полученных студентом за каждый компонент формируемых компетенций, с учетом результатов текущего и рубежного контроля в виде интегральной оценки по 4-х балльной шкале. Все результаты контроля заносятся в оценочный лист и заполняются преподавателем по итогам промежуточной аттестации.

Форма оценочного листа и требования к его заполнению приведены в общей части ФОС образовательной программы.

При формировании итоговой оценки промежуточной аттестации в форме экзамена используются типовые критерии, приведенные в общей части ФОС образовательной программы.

Тестовые задания для самопроверки