Раздел 1

1.Поясните разницу между ссылкой и указателем.

2.Поясните разницу между константным указателем на неконстантную переменную и неконстантным указателем на константную переменную.

3.Поясните разницу между классом и объектом.

4.Поясните разницу между равенством и идентичностью объектов

5.Дайте определение инкапсуляции, объясните её назначение и реализацию в C++

6.Дайте определение наследованию, объясните его назначение и реализацию в C++

7.Приведите типы наследования в C++, поясните разницу между ними

8.Объясните назначение конструктора копирования и привидите случаи, когда его необходимо определить

9.Дайте определение полиморфизму, объясните его назначение и реализацию в C++

10.Объясните назначение виртуальных и чисто виртуальных функций

11.Дайте определение абстрактного и полиморфного классов

12.Объяните назначение виртуальных деструкторов

13.Дайте определение статического и динамического полиморфизма

14.Нарушают ли функции-друзья принцип инкапсуляции?

15.Объяните назначение header guards

16.Перегрузка каких операторов допустима в С++?

17.Каким образом осуществляется доступ к членам класса из методов данного класса?

18.По какой причине некоторые операторы могут быть перегружены только в виде функций-друзей класса?

19.Какие существуют способы завершения программы в С++?

20.Что происходит, если сгенерированное исключение не было перехвачено?

21.Поясните принуцип RAII.

22.Какие существуют способы обработки ошибок в C++?

23.В чем состоит раница между rvalue- и lvalue-выражениями?

24.Пояните понятие rvalue-ссылок и способы их применения

25.В чем состоит разница между копированием и перемещением?

26.Какие оптимизации выполняют современные компиляторы C++?

27.Каким образом происходит компиляция шаблонных функций?

28.Каким образом происходит выводи типа шаблона?

29.Виды формальных параметров шаблона

Раздел 3

1.Дайте определение составного типа (aggregation). Каким требованиям должен удовлетворять тип, чтобы к нему была применима aggregate initialization?

2.Поясните разницу между отношениями композиции и отношениями агрегации.

3.Какие существуют способы оценки амортизированного времени выполнения операций?

4.Что такое аллокатор и каким образом он может использоваться?

5.В чем состоит назначение итераторов?

6.Как реализован итератор на конец вектора (массива)? В чем причины такой реализации?

7.По какой причине операции, уменьшающие размер вектора, не освобождают память?

8.Как реализована двусторонняя очередь?

9.Дайте определение односвязному и двусвязному списку.

10.Приведите алгоритмическую сложность операций с вектором, списком, двусторонней очередью.

11.По какой причине могут инвалидироваться итераторы? В чем состоит инвалидация итератора?

12.Как реализованы ассоциативные контейнеры в STL? Приведите и обоснуйте сложность операций с ними.

13.Как реализованы неупорядоченные контейнеры в STL? Приведите и обоснуйте сложность операций с ними.

14.Дайте определение хэш-функции, коллизии, универсальному хэшированию

15.Привидите примеры хэш-функций для целочисленных типов и для строк.

16.Как реализован std::unique\_ptr? Почему для его реализации необходима поддержка семантики перемещения?

17.Как реализован std::shared\_ptr?

Раздел 4

1.В чем состоит назначение метаобъектного компилятора фреймворка Qt?

2.Опишите иерархию классов фреймворка Qt.

3.Опишите назначение и принципы использования сигналов в фреймворке Qt.

Практические задания

Раздел 1

1.Реализуйте программу для подсчёта ежемесячных расходов студента с использованием ООП. Данные о расходах представлены в файлах в формате csv

2.В зоомагазине в стоящих в ряд клетках содержатся различные животные, в частности, кошки и собаки. Все животные имеют имя, возраст и породу, а также любят играть и имеют голоса. Кроме того, имеется возможность определить вид животного. В зоомагазин можно приносить животных из приюта и забирать их домой.

Каждый класс-животное должен:

●хранить имя, возраст, породу и вид животного;

●иметь чисто виртуальные функции-члены say() и play(), которые выводят в консоль голос животного и сообщение о его игре;

●иметь функцию-член getSpicies(), возвращающую вид животного;

●определять оператор вставки в поток, выводящий сообщение о животном: его имени, виде, породе и возрасте.

Класс-зоомагазин должен:

●обеспечивать возможность хранения любого количества животных;

●иметь функцию get(), освобождающую последнюю занятую клетку и возвращающую указатель на содержавшийся в ней объект-животное. Объяснить, почему данное решение является потенциальным источником проблем;

●иметь функцию put(Pet\*), позволяющую помещать новых животных в первую свободную клетку;

●определять оператор индексирования, обеспечивающий доступ к клеткам с животными по их индексу в массиве;

●определять оператор копирующего присваивания и конструктор копирования;

●определять оператор вставки в поток, помещающий животное в первую свободную клетку;

●определять оператор извлечения из потока, извлекающий животное из последней занятой клетки;

●определять оператор сравнения зоомагазинов: если в клетках содержатся одинаковые животные, значит, это тот же зоомагазин;

●определять оператор вызова функции, выводящий в консоль справку о названии зоомагазина, содержащихся в нём животных и количестве свободных клеток.

Раздел 3

1.Реализовать программу для проверки сбалансированности скобок в тексте программного кода и вывести позицию первой несбалансированной скобки

2.Реализовать программу для подсчёта непрерывных числовых последовательностей в строке

3.Реализовать сортирующее дерево (std::priority\_queue)

4.Реализовать std::deque и итератор для неё таким образом, чтобы она была совместима с алгоритмами STL

Раздел 4

1.Реализовать графический интерфейс для задачи 2 раздела 1 с использованием фреймворка Qt

Вопросы для защиты практических работ

Раздел 1

1.Дайте определение стандартным потокам ввода, вывода, ошибок. Каким образом реализована работа с ними в С++?

2.Дайте определение потоку.

Поясните разницу между выделением памяти на стеке и на куче.

3.В чем назначение ключевого слова const?

4.Поясните разницу между передачей параметра по ссылке, по константной ссылке и по значению. В каких случаях следует применять каждый из данных подходов?

5.Поясните разницу между статическими и нестатическими методами и свойствами классов.

6.Поясните разницу между константными и неконстантными методами классов.

7.Что представляет собой std::string? Каковы вероятные реализации std::string?

8.Каковы вероятные причины запрета на перегрузку некоторых операторов?

Раздел 3

1.Какие сущесвуют виды итераторов в C++? Чем обусловлена необходимость в введении различных типов итераторов?

2.Дайте определение лямбда-выражению. Пояните синтаксис лямбда-выражений в C++.

3.Какой тип имеют лямбда-выражения в C++?

4.В чём назначение ключевого слова auto? Как происходит вывод типа?

5.В чём назначение ключевого слова constexpr?

6.Что такое члены типов класса? Как используются члены типов при работе с STL?

7.Как сделать пользовательский тип данных совместимым с алгоритмами STL?

Раздел 4

1.Как скомпиировать программу, использующую фреймворк Qt?

2.Каким образом фреймворк Qt расширяет множество зарезервированных ключевых слов C++?

Оценочные материалы для промежуточной аттестации

Задачи к экзамену

1.Реализовать класс матрицы целых чисел. Размерность матриц не ограничена. Класс должен:

●считывать элементы матрицы из файла формата csv;

●реализовывать конструкторы копирования и перемещения (если необходимо);

●реализовывать операторы копирующего и перемещающего присваивания (если необходимо);

●реализовывать операторы сложения, вычитания, умножения матриц;

●обрабатывать ошибки с использованием исключений

2.Реализовать класс матрицы целых чисел. Размерность матриц не ограничена. Класс должен:

●считывать элементы матрицы из файла формата csv;

●реализовывать конструкторы копирования и перемещения (если необходимо);

●реализовывать операторы копирующего и перемещающего присваивания (если необходимо);

●реализовывать операторы сложения, вычитания, сравнения матриц, умножения матрицы и числа;

●обрабатывать ошибки с использованием исключений

3.Реализовать шаблонный класс матрицы. Размерность матриц не ограничена. Класс должен:

●реализовывать методы для заполнения матрицы поэлементно и построчно;

●реализовывать операторы сложения и сравнения матриц (==, !=);

●реализовывать конструкторы копирования и перемещения (если необходимо);

●реализовывать операторы копирующего и перемещающего присваивания (если необходимо);

●реализовывать оператор выставки в поток (<<) для вывода матрицы в консоль;

●обрабатывать ошибки с использованием исключений

4.Реализовать шаблонный класс матрицы. Размерность матриц не ограничена. Класс должен:

●реализовывать методы для заполнения матрицы поэлементно и построчно;

●реализовывать оператор индексирования;

●реализовывать конструкторы копирования и перемещения (если необходимо);

●реализовывать операторы копирующего и перемещающего присваивания (если необходимо);

●реализовывать метод, транспонирующий матрицу;

●реализовывать оператор выставки в поток (<<) для вывода матрицы в консоль;

●обрабатывать ошибки с использованием исключений.

5.Разработать класс гистограммы целочисленных значений. Класс должен:

●определять конструктор, принимающий массив значений и количество интервалов. Диапазон значений разбивается на данное число интервалов и подсчитывается количество значений, попавших в каждый интервал. Совокупность шкалы интервалов и подсчитанных таким образом значений и есть гистограмма;

●определять конструктор по умолчанию;

●определять конструкторы копирования и перемещения (если необходимо);

●реализовывать операторы копирующего и перемещающего присваивания (если необходимо);

●определять оператор вставки в поток (<<), выводящий гистограмму в консоль при помощи ранее заданного символа. Пример вывода:

+++

+++++++

++

+++

●обрабатывать ошибки при помощи механизма исключений

6.Cмоделировать стрельбу по мишени группой человек. Каждый человек имеет свое имя, возраст (в годах) и стаж обучения стрельбе (в годах). Люди делятся на новичков, опытных и ветеранов. Каждый класс-человек должен реализовывать метод «shoot» без параметров, возвращающий логическое значение (попал – true, не попал – false). Факт попадания определяется случайным образом, причем для новичка вероятность попасть равна 0,01\*стаж обучения; для опытного = 0,05\*стаж обучения стрельбе; для ветерана = 0,9 – 0,01\*возраст. Люди стреляют по очереди, начиная с первого, пока кто-то не попадет в мишень. Стрельба прекращается после того, как кто-то попал или все выстрелили по одному разу. После каждого выстрела необходимо выводить на экран всю информацию о стрелявшем (перегрузить оператор вставки в поток) и результат стрельбы.

7.Смоделировать сдачу зачета студентами. Каждый студент характеризуется ФИО и количеством посещенных занятий. Студенты делятся на обычных, сообразительных и гениев. Каждый студент должен иметь метод «Сдать зачет» без параметров, возвращающий логическое значение (сдал – true, не сдал – false). Обычные студенты точно сдают зачет, если посетили все занятия, если были более чем на половине занятий, то сдают с вероятностью 0,5; иначе не сдают. Сообразительные студенты тоже точно сдают зачет, если посетили все занятия, если были более чем на половине занятий, то сдают с вероятностью 0,7; иначе не сдают зачет. Гении точно сдают зачет, если были хотя бы на одном занятии, иначе не сдают. Создайте массив из произвольного числа различных студентов, и смоделируйте сдачу зачета с выводом подробных результатов всех сведений о студентах (перегрузить оператор вставки в поток), а также результате сдачи.

8.Смоделировать нарезку листов стали. Листы стали характеризуются толщиной. Листы делятся на квадратные, прямоугольные и треугольные (в виде прямоугольного треугольника, задаются двумя катетами, в мм). Для каждый тип стального листа должен иметь метод «Площадь», возвращающий площадь листа. Также определите метод «Вес», который вычисляет вес листа, умножая площадь листа на его толщину и плотность стали. В главной программе создайте массив из произвольного числа различных листов стали. Выведите информацию о листах (перегрузить оператор вставки в поток) и рассчитайте суммарную площадь и суммарный вес всех листов.

9.Разработать шаблонный класс односвязного списка. Класс должен:

●определять метод для добавления элемента в начало списка;

●определять метод для удаления элемента из начала списка;

●определять метод для проверки, пустой ли список;

●определять оператор сложения списков, добавляющий второй список в конец первого;

●определять операторы постфиксного и префиксного декрементов, удаляющие элемент из начала списка;

●определять конструкторы копирования и перемещения (если необходимо);

●определять операторы копирующего и перемещающего присваивания (если необходимо);

●обрабатывать ошибки при помощи механизма исключений

10.Разработать шаблонный класс односвязного списка. Класс должен:

●определять метод для добавления элемента в начало списка;

●определять метод для добавления элемента в конец списка;

●определять метод для удаления элемента из начала списка;

●определять метод для проверки, пустой ли список;

●определять оператор умножения списка на число, повторяющий данный список N раз (пример: [1, 2, 3] \* 2 = [1, 2, 3, 1, 2, 3]);

●определять оператор вставки в поток, выводящий список в консоль;

●определять конструкторы копирования и перемещения (если необходимо);

●определять операторы копирующего и перемещающего присваивания (если необходимо);

●обрабатывать ошибки при помощи механизма исключений

11.Разработать шаблонный класс односвязного списка. Класс должен:

●определять метод для добавления элемента в начало списка;

●определять метод для удаления элемента из начала списка;

●определять метод для проверки, пустой ли список;

●определять метод join, возвращающий строку с перечислением элементов списка через разделитель, указанный в параметре метода (пример: [1, 2, 3].join(‘,’) = “1, 2, 3”);

●определять конструктор, принимающий строку с перечислением элементов списка через разделитель, и данный разделитель, как показано в примере выше. Принять, что ошибка преобразования типов невозможна, однако, формат строки может быть некорректным;

●определять конструкторы копирования и перемещения;

●определять операторы копирующего и перемещающего присваивания;

●обрабатывать ошибки при помощи механизма исключений

12.Разработать шаблонный класс комплексного числа в алгебраической форме. Принять, что пользователь не может задать невалидный формальный параметр шаблона (например, строку). Класс должен:

●содержать реальную и мнимую части и методы для их задания и доступа к ним;

●определять операторы сложения, вычитания, умножения, деления, сравнения комплексных чисел (==,!=);

●определять оператор умножения данного числа на действительное число;

●определять метод, возвращающий сопряжённое число;

●определять метод, возвращающий модуль комплексного числа;

●определять оператор вставки в поток (<<), выводящий в консоль комплексное число в виде Re при наличии только реальной части, iIm при наличии только мнимой части, Re + iIm в противном случае;

●обрабатывать ошибки при помощи механизма исключений

13.Разработать шаблонный класс комплексного числа в тригонометрической форме. Принять, что пользователь не может задать невалидный формальный параметр шаблона (например, строку). Класс должен:

●содержать модуль и аргумент числа и методы для их задания и доступа к ним;

●определять операторы умножения, деления, сравнения комплексных чисел (==, !=);

●определять оператор умножения данного числа на действительное число;

●определять метод для возведения данного числа в степень N;

●определять метод, возвращающий сопряжённое число;

●определять методы возвращающие действительную и мнимую части данного числа в алгебраической форме;

●определять оператор вставки в поток (<<), выводящий в консоль комплексное число в виде z(cos(f) + i\*sin(f));

●обрабатывать ошибки при помощи механизма исключений

14.Разработать шаблонный класс Pair, хранящий пару значений разных типов. Класс должен:

●реализовывать методы для доступа к значениям и задания значений;

●реализовывать операторы сравнения (==, !=);

●реализовывать оператор сложения, складывающий соответствующие элементы пары;

●реализовывать оператор выставки в поток (<<), выводящий в консоль пару в виде “(first, second)”;

●реализовать оператор извлечения из потока (>>), принимающий два значения из консоли и заполняющий ими пару строк;

●метод, принимающий два значения и возвращающий составленную из них пару;

●метод, принимающий пару и возвращающий пару, в которой первое значение поставлено на место второго, второе – на место первого

Задание на курсовую работу

Необходимо выбрать игру с несложным графическим интерфейсом и согласовать выбор с преподавателем. Примеры: змейка, тетрис, сапер. Реализовать приложение, предоставить исходный код и составить пояснительную записку, содержащую:

Введение

1. Техническое задание

1.1. Введение

1.1.1. Наименование программы

1.1.2. Назначение программы

1.2. Область применения программы

1.3. Обоснование разработки

1.4. Тебования к программе:

1.4.1. Требования к функциональным характеристикам программы

1.4.2. Требования к техническим средствам, используемым при работе программы

1.4.3. Требования к языкам программы и среде разработки программы

1.5. Требования к программной документации

1.6. Этапы разработки

2. Обзор способов организации данных и обоснование выбора структуры данных для эффективного выполнения операций

3. Описание программы

3.1. Общие сведения

3.1.1. Наименование программы

3.1.2. Программное обеспечение, необходимое для функционирования программы

3.1.3. Язык программирования, на котором написана программа

3.2. Функциональное назначение программы

3.3 Описание логической структуры программы:

3.3.1 Алгоритмы, используемые в программе

3.3.2 Структура программы с описанием функций составных частей и связей между ними

3.4 Технические средства

3.5 Вызов программы

Заключение