

Задача «Матрица»

(перегрузка операций и функций)

Запрограммируйте класс `M`, с объектами которого можно выполнить следующие операции:

```
M m; //создание матрицы 3x3 с нулевыми элементами
M n(2,2); // создание матрицы 2x2 с нулевыми элементами
M o(1); //создание матрицы 2x2 с единичной диагональю
M p("1 2 3 4 5 6 7 8 9"); //создание квадратной матрицы,
//значения элементов 0...9 разделяются пробелами
M q = m + p; //сложение двух матриц
//и помещение результата в третью
n - o; //вычитание матриц
5 + n; //добавление константы ко всем элементам матрицы
n - 5; //вычитание константы из всех элементов матрицы
```

Задача «Планировщик»

(файловые потоки)

Обеспечьте ввод времени (ЧЧ:ММ) начала и окончания, а также наименование (от 1 до 30 символов) задачи из текстового файла. Выведите на экран отметки времени от 07:00 до 23:00 с шагом 15 минут и отметьте символом «*» промежутки времени, которые заняты хотя бы одним заданием.

Задача «Резьба»

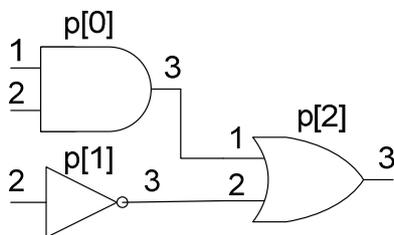
(наследование)

Резьбовое соединение заданного диаметра состоит из одного винта заданной длины, нескольких шайб и гаек. Выделите общие свойства и методы перечисленных деталей в виртуальный базовый класс и предоставьте интерфейс для создания/модификации соединений.

Задача «Логика»

(наследование)

Логическую схему можно собрать, используя 3 типа элементов: AND, OR, NOT. Первые 2 типа элементов имеют по 2 входа и один выход, а третий – только один вход (предлагается игнорировать 1-й вход элемента NOT, тогда второй и третий выходы будут аналогичны другим элементам). Выделите общие свойства и методы деталей в виртуальный базовый класс и предоставьте интерфейс для создания/модификации соединений. Например, для приведенной ниже схемы приведен примерный код C++ (pin – вывод элемента):



```
//создание элементов:
```

```
p[0] = new AND;
```

```
p[1] = new NOT;
```

```
p[2] = new OR;
```

```
//создание соединений:
```

```
Connect(p[2]->pin1, p[0]->pin3);
```

```
Connect(p[2]->pin2, p[1]->pin3);
```

```
//установка значений на входах:
```

```
Set(p[0],1,true);
```

```
Set(p[0],2,false);
```

```
Set(p[1],2,true);
```

```
//вычисление
```

```
bool result = Calculate(p[2],3);
```

Задача «Фигуры»

(наследование)

Даны несколько геометрических фигур – линия, треугольник и квадрат. Выделите общие свойства и методы фигур в виртуальный базовый класс и предоставьте интерфейс для вращения каждой из фигур относительно заданной точки. Отобразите фигуры в текстовом виде.

Задача «Записи»

(бинарные файлы)

Бинарный файл содержит записи с полями: дата и время ввода, наименование и числовой параметр. Обеспечьте функциональность добавления, удаления, поиска/замены и сортировки по одному из трех полей.

Задача «multimap»

(STL)

С помощью шаблона `multimap` создайте редактируемый русско-английский словарь. Обеспечьте его работу в двухстороннем порядке (русско-английский и англо-русский) с предсказанием ввода: после ввода очередного символа с клавиатуры выводится подсказка из начинающихся с введенной подстроки словарных слов (до 10-ти).

Задача «map»

(STL)

С помощью шаблона `map` создайте связи между номерами выводов микросхемы и их названиями. Кроме наименования, каждый вывод имеет одну из категорий: питание, аналоговый вход, аналоговый выход, цифровой вход, цифровой выход, аналоговый вход-выход, цифровой вход-выход, не подключено. Обеспечьте вывод наименований в алфавитном порядке внутри категорий и соответствующих номеров выводов. Предусмотрите вывод номеров выводов в порядке возрастания с указанием соответствующих наименований.

Задача «Пул»

(перегрузка операций)

Пул ресурсов – конечный набор извлекаемых и возвращаемых обратно ресурсов, например единиц спортивного инвентаря в прокате. Создайте несколько редактируемых пулов, редактируемый перечень пользователей таких пулов. Обеспечьте функции администрирования для обслуживания пользователей, пополнения и обновления пулов. Операция для помещения – `operator << ()`, для извлечения – `operator >> ()`.

Задача «Тэги»

(файловые потоки + STL)

Разработайте интерфейс для работы с файлами по их тэгам. Тэг – ключевое слово, описывающее содержимое файла (фото, отчет, курсовая, 2020 г.). Тэги хранятся в отдельном текстовом файле, каждый в своей строке. Информация о связи файлов и тэгов хранится в бинарном файле. Обеспечьте возможность ввода новых тэгов, привязки тэгов к файлу, выявлению файлов без тэгов, вывод файлов с заданными тэгами.

Имеется функция `myGetFileNames()`, входом которой является строка с именем папки, а выходом – список `list` с полными именами всех хранящихся в заданной папке файлов.