

Лабораторная работа №8

«Разработка шаблона класса»

Скоробогатов С.Ю.

4 мая 2018 г.

1 Цель работы

Целью данной работы является изучение шаблонов классов языка C++.

2 Исходные данные

2.1 Некоторые контейнерные классы

В некоторых вариантах заданий размер данных, хранящихся в объекте класса, не фиксирован, и, тем самым, требуется динамическое выделение памяти. Чтобы избежать реализации конструктора копий, деструктора и перегруженной операции присваивания, целесообразно в своём классе организовать хранение данных в объектах контейнерных классов из стандартной библиотеки C++.

Динамические массивы в C++ представляются переменными класса `vector`, объявленного в заголовочном файле `vector`. Использование этого класса можно проиллюстрировать следующим примером (создание вектора целых чисел и заполнение его числами от 0 до 9):

```
vector<int> a;  
for (int i = 0; i < 10; i++) a.push_back(i);
```

К вектору можно применять операцию индексации. Размер вектора возвращает метод `size`. Поэтому, например, вывести элементы нашего вектора на печать можно следующим образом:

```
for (int i = 0; i < a.size(); i++) cout << a[i];
```

Вектора, кроме того, предоставляют итераторы для доступа к своему содержимому. Это даёт возможность для перебора элементов вектора использовать специальную форму оператора `for`:

```
for (int x : a) cout << x;
```

Упорядоченный ассоциативный массив, реализованный через дерево, представлен в стандартной библиотеке C++ шаблоном класса `map` из заголовочного файла `map`.

Следующий фрагмент кода демонстрирует объявление ассоциативного массива, заполнение его значениями и вывод словарных пар на печать:

```
map<string , int> m;
m["a"] = 1;  m["b"] = 2;
for (auto pair : m)
    cout << pair.first << ", " << pair.second << endl;
```

Обратите внимание на то, что применение ключевого слова `auto` в объявлении переменной `pair` заставляет компилятор выводить её тип автоматически. Эта возможность доступна только в режиме `c++11`.

Упорядоченное множество, также реализованное через дерево, представлено шаблоном класса `set` из заголовочного файла `set`. Объявление множества, добавление в него элементов и вывод множества на печать можно проиллюстрировать следующим фрагментом кода:

```
set<string> s;
s.insert("qwerty");
s.insert("abcd");
for (string x : s) cout << x << endl;
```

Ассоциативный массив и множество, реализованные через хеш-таблицу, представлены в стандартной библиотеке языка C++ шаблонами классов `unordered_map` и `unordered_set`, соответственно.

2.2 Специализация шаблонов в зависимости от значений их целочисленных параметров

Информация об объявлении и специализации шаблонов дана в лекционном курсе. Здесь мы рассмотрим только один вопрос, касающийся специализации шаблонов в зависимости от значений их целочисленных параметров.

Пусть шаблон некоторого класса `Abs` имеет целочисленный параметр `N`:

```
1  template <int N>
2  class Abs
3  {
4  public:
5      void print();
6  };
7
8  template <int N>
9  void Abs<N>::print()
10 {
11     cout << N << endl;
12 }
```

Допустим, мы хотим добиться, чтобы в этом совершенно искусственном примере метод `print` всегда печатал абсолютное значение числа `N`. При этом мы собираемся достигнуть этого результата через специализацию шаблона класса.

Добавим в шаблон дополнительный булевский параметр `isNegative` со значением по умолчанию, вычисляемым на основе значения параметра `N`:

```
1  template <int N, bool isNegative = (N > 0)>
2  class Abs
```

```

3 {
4 public:
5     void print();
6 };
7
8 template <int N, bool isNegative>
9 void Abs<N, isNegative>::print()
10 {
11     cout << N << endl;
12 }

```

Очевидно, что при отрицательных значениях параметра N значение параметра `isNegative` будет истинным. Поэтому мы можем добавить специализированную версию шаблона для отрицательных N :

```

14 template <int N>
15 class Abs<N, true>
16 {
17 public:
18     void print();
19 };
20
21 template <int N>
22 void Abs<N, true>::print()
23 {
24     cout << -N << endl;
25 }

```

Теперь при инстанцииции шаблона класса `Abs` с отрицательным N компилятор будет действовать специализированную версию, в которой N печатается с изменением знака. В этом можно убедиться, добавив в нашу программу следующий код:

```

27 int main()
28 {
29     Abs<-5> t;
30     t.print();
31     return 0;
32 }

```

3 Задание

Согласно выбранному из таблиц 1–16 описанию требуется составить шаблон класса, разместив его в отдельном заголовочном файле. Проверку работоспособности класса требуется организовать в функции `main`, размещённой в файле «`main.cpp`».

Таблица 1: Варианты классов

№	Описание	Студент	Группа	Дата
1	FenwickTree<T, N> – дерево Фенвика для последовательности длины N с элементами типа T , имеющее операцию вычисления исключающего ИЛИ элементов на заданном отрезке и операцию изменения указанного элемента. Дерево должно быть реализовано через массив размера N . В FenwickTree< bool , N> каждому элементу должен соответствовать 1 бит.	Богданова	ИУ9И-21	04.05
2	SparseTable<T, N> – разреженная таблица для последовательности длины N с элементами типа T , имеющая операцию вычисления максимума на заданном отрезке. Таблица должна быть представлена в виде двумерного массива с константными размерами, вычисляемыми компилятором во время компиляции.	Бакланов	ИУ9-21	04.05
3	Diraph< int N, bool Dense> – простой ориентированный граф, у которого N вершин и про который известно, будет ли он плотным или разреженным. Плотный граф должен быть реализован через матрицу смежности, а разреженный – через списки инцидентности. В графе должны быть реализованы операции: добавление дуги (принимает номера вершин, которые надо соединить) и проверка смежности двух вершин.	Боровик	ИУ9-21	04.05
4	RangeSet<T, Closed> – набор интервалов, имеющих вид (a, b) при Closed== false или $[a, b]$ при Closed== true . Набор интервалов задаёт множество чисел типа T . Операции: проверка принадлежности числа множеству; добавление нового интервала. Если тип T – целочисленный, в классе должна быть дополнительная операция, вычисляющая количество чисел, принадлежащих задаваемому набором интервалов множеству.	Громков	ИУ9-21	04.05
5	Points<T> – множество точек на плоскости, в которой каждая точка помечена значением типа T . Операции: добавление новой точки; удаление точек, попадающих внутрь указанного круга. Если T – числовой тип, то метка точки должна трактоваться как её масса, и в классе должна быть дополнительная операция, вычисляющая центр масс множества точек.	Дмитриев	ИУ9-21	04.05

Таблица 2: Варианты классов

№	Описание	Студент	Группа	Дата
6	Path<T, M> – траектория движения точки в M -мерном пространстве, заданная с помощью n положений точки в моменты времени $0, 1, \dots, n - 1$. Координаты точки задаются числами типа T . Операции: вычисление длины траектории; добавление нового положения точки в конец траектории. В случае, если $M = 1$, в классе должна быть дополнительная операция, вычисляющая минимальную и максимальную координату точки на траектории.	Егорычев	ИУ9-21	04.05
7	Matrix<M,N> – целочисленная матрица размера $M \times N$ с операцией, возвращающей ссылку на указанный элемент. Если $M = N$, то для матрицы должна быть доступна операция возведения в квадрат.	Кочанова	ИУ9-21	04.05
8	Equality<L, H> – линейное равенство вида $a_1x_1 + a_2x_2 + \dots + a_nx_n = b$, коэффициенты a_i и b которого заданы целыми числами, лежащими в диапазоне от L до H . В классе должна быть предусмотрена операция проверки, удовлетворяет ли указанный вектор значений переменных равенству. Коэффициенты равенства должны храниться в виде $a_i - L$, причём для их представления должен использоваться целочисленный тип минимального размера, подходящий для представления числа $H - L + 1$.	Кузнецов	ИУ9-21	04.05
9	PascalArray<int L,int R,class T> – массив с элементами типа T , индексируемыми от L до R , с перегруженной операцией индексации и операцией конкатенации двух массивов, которая допустима в случае, если правая граница первого массива на единицу меньше левой границы второго массива. Массив, в котором $R = L - 1$, считается пустым и не обладает операцией индексации.	Лобачев	ИУ9-21	04.05
10	Queue<class T,int N> – очередь с элементами типа T и максимальным размером N , который может быть не задан (равен 0). Для очереди должны быть реализованы операции enqueue, dequeue и is_empty. Очередь с ненулевым максимальным размером должна быть реализована через массив размера N , хранящийся в поле объекта очереди.	Маркова	ИУ9-21	04.05

Таблица 3: Варианты классов

№	Описание	Студент	Группа	Дата
11	<p>Polygon<T, N> – N-угольник на плоскости, заданный координатами вершин. Координаты вершин представлены числами типа T.</p> <p>Операции: вычисление периметра (возвращает double); добавление новой вершины (в результате формируется новый (N + 1)-угольник).</p> <p>Если N = 3, в классе должна быть дополнительная операция вычисления площади треугольника (возвращает double).</p>	Петрова	ИУ9-21	04.05
12	<p>Seq<class T, bool Unique> – последовательность с элементами типа T, которая может допускать или не допускать наличие повторяющихся элементов. Для последовательности должна быть перегружена операция индексации (она должна возвращать константную ссылку в случае последовательности с уникальными элементами, чтобы элемент по ссылке нельзя было изменить).</p> <p>Кроме того, должна быть реализована операция добавления нового элемента в последовательность.</p>	Поленов	ИУ9-21	04.05
13	<p>IntStack<L, H> – стек целых чисел из диапазона от L до H, имеющий стандартный для стека набор операций.</p> <p>Если размер диапазона не превышает 256, для представления стека должен использоваться массив char'ов. В случае диапазона, имеющего размер, не превышающий 65536, должен использоваться массив short'ов.</p>	Родионов	ИУ9-21	04.05
14	<p>Polynom<int L, int H, int N> – полином степени N с элементами из диапазона от L до H, имеющий следующие операции:</p> <p>дифференцирование полинома, в результате которого должен формироваться новый полином типа Polynom<L2, H2, N2>, где L2, H2 и N2 вычисляются на базе значений L, H и N;</p> <p>вычисление значения полинома в точке.</p> <p>Если размер диапазона не превышает 256, для представления полинома должен использоваться массив char'ов.</p>	Санталов	ИУ9-21	04.05

Таблица 4: Варианты классов

№	Описание	Студент	Группа	Дата
15	<p>Polygon<T> – многоугольник на плоскости, заданный координатами вершин. Координаты вершин представлены числами типа T.</p> <p>Операции: вычисление периметра (возвращает double); добавление новой вершины.</p> <p>В случае, если T – double, в классе должна быть дополнительная операция поворота N-угольника относительно его первой вершины на такой угол, чтобы его первое ребро стало параллельно оси OX.</p>	Свечникова	ИУ9-21	04.05
16	<p>Matrix<T,N> – квадратная матрица размера N, элементы которой имеют тип T.</p> <p>Матрица должна иметь следующие операции:</p> <ol style="list-style-type: none"> 1. запись значения в элемент с индексами (i, j); 2. чтение значения из элемента с индексами (i, j); 3. построение новой матрицы путём удаления i-той строки и j-того столбца. <p>В Matrix<bool,N> при $N \leq 8$ матрица должна быть представлена 64-разрядным целым числом, в котором каждому элементу соответствует один бит.</p>	Тарасова	ИУ9-21	04.05
17	<p>Matrix<L, H, M, N> – целочисленная матрица размера $M \times N$, элементы которой принадлежат диапазону от L до H.</p> <p>Матрица должна иметь следующие операции:</p> <ol style="list-style-type: none"> 1. запись значения в элемент с индексами (i, j); 2. чтение значения из элемента с индексами (i, j); 3. транспонирование. <p>Если размер диапазона не превышает 256, для представления матрицы должен использоваться массив char'ов.</p>	Узунов	ИУ9И-21	04.05

Таблица 5: Варианты классов

№	Описание	Студент	Группа	Дата
18	<p>$\text{Stack}\langle T \rangle$ – стек с элементами типа T, имеющий в дополнение к обычным стековым операциям операцию переворачивания стека. Все операции должны работать за константное время.</p> <p>Для представления стека нужно использовать двунаправленный список.</p> <p>В $\text{Stack}\langle \text{int} \rangle$ дополнительно должна быть реализована операция, сообщающая о наличии в стеке нулевого элемента и работающая за константное время.</p>	Филоненко	ИУ9-21	04.05
19	<p>$\text{SegmentTree}\langle T, N \rangle$ – дерево отрезков для последовательности длины N с элементами типа T, имеющее операцию вычисления суммы элементов на заданном отрезке и операцию изменения указанного элемента.</p> <p>Отметим, что в случае, если элементами последовательности являются строки, от дерева отрезков нет никакой пользы, так как всё равно сумма строк вычисляется за время, пропорциональное длине результирующей строки. Соответственно, $\text{SegmentTree}\langle \text{string}, N \rangle$ должно быть реализовано через обычный массив строк размера N.</p>	Шатнюк	ИУ9-21	04.05
20	<p>$\text{Seq}\langle \text{class } T, \text{bool Sorted} \rangle$ – последовательность с элементами типа T, которая может быть отсортирована или не отсортирована. Для последовательности должна быть перегружена операция индексации (она должна возвращать константную ссылку в случае отсортированной последовательности, чтобы элемент по ссылке нельзя было изменить). Кроме того, должны быть операции добавления и удаления элемента последовательности.</p>	Шельдяев	ИУ9-21	04.05
21	<p>$\text{Stack}\langle \text{class } T, \text{int } N \rangle$ – стек с элементами типа T и максимальным размером N, который может быть не задан (равен 0). Для стека должны быть реализованы обычные стековые операции. Стек с ненулевым максимальным размером должен быть реализован через массив размера N, хранящийся в поле объекта стека.</p>	Ярахмедов	ИУ9-21	04.05

Таблица 6: Варианты классов

№	Описание	Студент	Группа	Дата
22	<p>Queue<T, N> – «неизменяемая» очередь с элементами типа <i>T</i> и максимальным размером <i>N</i>, имеющая обычные для очереди операции.</p> <p>«Неизменяемость» очереди заключается в том, что операции Enqueue и Dequeue вместо изменения очереди, для которой они вызваны, создают и возвращают новую очередь, отличающуюся от исходной на один элемент. При этом исходная очередь полностью сохраняет своё состояние и работоспособность.</p> <p>Операции Enqueue и Dequeue нужно реализовать так, чтобы они работали за амортизированное константное время. Для этого нужно использовать два стека размера <i>N</i>. Операция Enqueue записывает новый элемент в первый стек, а операция Dequeue забирает элемент из второго стека. Фокус заключается в том, что стеки выделяются в динамической памяти и являются общими для очереди, на которой сработала операция, и для очереди, которая была создана в результате работы операции. Отметим, что новая пара стеков создаётся в момент создания новой пустой очереди, а также тогда, когда операция Dequeue обнаруживает, что второй стек пуст и нужно копировать содержимое первого стека во второй.</p> <p>Для освобождения памяти, занимаемой парой стеков, которая потенциально может разделяться сразу несколькими очередями, требуется воспользоваться подсчётом ссылок на эту пару стеков (можно воспользоваться шаблоном <code>shared_ptr</code>).</p>	Апахов	ИУ9-22	04.05
23	<p>List<T> – последовательность значений типа <i>T</i>, реализованная через двунаправленный список с операциями поиска значения, вставки значения в указанное место, удаления значения.</p> <p>В List<int> дополнительно должна присутствовать операция вычисления суммы значений.</p>	Бахметьев	ИУ9-22	04.05

Таблица 7: Варианты классов

№	Описание	Студент	Группа	Дата
24	<p>$\text{Matrix}\langle T, M, N \rangle$ – матрица размера $M \times N$, элементы которой имеют тип T. Матрица должна иметь следующие операции:</p> <ol style="list-style-type: none"> 1. запись значения в элемент с индексами (i, j); 2. чтение значения из элемента с индексами (i, j); 3. транспонирование. <p>В $\text{Matrix}\langle \text{bool}, M, N \rangle$ при $M \leq 8$, $N \leq 8$ матрица должна быть представлена 64-разрядным целым числом, в котором каждому элементу соответствует один бит.</p>	Бойчук	ИУ9-22	04.05
25	<p>$\text{Seq}\langle T, N \rangle$ – неизменяемая упорядоченная последовательность длины N элементов типа T, имеющая операцию поиска первого вхождения в неё подпоследовательности длины M. В операции поиска должен использоваться алгоритм Кнута–Морриса–Пратта. Операция поиска должна создаваться шаблоном, имеющим параметр M. Этот шаблон должен быть специализирован для случая $M = N$, при котором алгоритм Кнута–Морриса–Пратта не нужен, а достаточно просто сравнить две последовательности.</p>	Ботвинников	ИУ9-22	04.05
26	<p>$\text{Stack}\langle T \rangle$ – стек с элементами типа T, имеющий в дополнение к обычным стековым операциям операцию вычисления максимального элемента, работающую за константное время. $\text{Stack}\langle \text{string} \rangle$ должен дополнительно иметь операцию переворачивания всех строк, находящихся в стеке, работающую за константное время.</p>	Браславский	ИУ9-22	04.05
27	<p>$\text{Polynom}\langle T, N \rangle$ – полином порядка N с коэффициентами типа T, имеющий операцию вычисления значения. В $\text{Polynom}\langle \text{bool}, N \rangle$ i-ый коэффициент означает наличие или отсутствие в полиноме члена x^i.</p>	Гавриленко	ИУ9-22	04.05

Таблица 8: Варианты классов

№	Описание	Студент	Группа	Дата
28	Operation< char O, class L, class R> – некоторое действие над двумя операндами. Операнды имеют типы L и R , каждый из которых – либо Operation, либо int (обозначает константу), либо int& (обозначает переменную). Константа O задаёт смысл действия: '+' – сложение, '*' – умножение, '=' – присваивание. В классе надо реализовать операцию выполнения действия.	Гавриловский	ИУ9-22	04.05
29	Stack< T , N > – стек с элементами типа T и максимальным размером N , имеющий обычные стековые операции. Стек должен быть реализован через массив размера N . В Stack< bool , N > каждый элемент должен быть представлен одним битом.	Гулин	ИУ9-22	04.05
30	Queue< T , N > – очередь с элементами типа T и максимальным размером N , реализованная через двойной стек и имеющая в дополнение к обычным для очереди операциям операцию переворачивания очереди, работающую за константное время. В Queue< int , N > дополнительно должна быть реализована операция вычисления суммы элементов очереди, работающая за константное время.	Даровская	ИУ9-22	04.05
31	List< T , N > – однонаправленный список, в элементах которого хранятся значения типа T , имеющий максимальный размер N . Размер N может быть не задан ($N = 0$), и тогда для представления списка нужно использовать динамически созданные структуры, связанные с помощью указателей. Если же размер N задан, то список должен быть представлен массивом. Для списка нужно реализовать операции: добавление элемента в начало списка, удаление первого элемента списка, поиск элемента по значению (возвращает bool), вычисление длины списка.	Испирян	ИУ9-22	04.05
32	Vector< T , N > – вектор размера N с элементами типа T , имеющий операции сложения, скалярного умножения и умножения на число. Vector< T , 3> дополнительно имеет операцию векторного умножения.	Климова	ИУ9-22	04.05

Таблица 9: Варианты классов

№	Описание	Студент	Группа	Дата
33	<p>$\text{Polynom}\langle T, N \rangle$ – полином порядка N с коэффициентами типа T, имеющий операции вычисления значения, сложения и дифференцирования.</p> <p>$\text{Polynom}\langle T, 2 \rangle$ и $\text{Polynom}\langle T, 1 \rangle$ дополнительно имеют операцию нахождения корней.</p>	Котова	ИУ9-22	04.05
34	<p>$\text{RQueue}\langle T, N \rangle$ – очередь с приоритетом максимального размера N с элементами типа T, имеющая одновременно операцию удаления максимального элемента и операцию удаления минимального элемента. Указанные операции должны работать за логарифмическое время.</p> <p>В $\text{RQueue}\langle \text{bool}, N \rangle$ все операции должны работать за константное время.</p>	Мамаев	ИУ9-22	04.05
35	<p>$\text{Polygon}\langle T, N \rangle$ – N-угольник на плоскости, заданный координатами вершин. Координаты вершин представлены числами типа T.</p> <p>Операции: вычисление периметра (возвращает <code>double</code>); вставка новой вершины после i-той вершины (в результате формируется новый $(N + 1)$-угольник).</p> <p>В случае, если T – <code>double</code>, в классе должна быть дополнительная операция поворота N-угольника относительно его первой вершины на заданный угол.</p>	Мирзоева	ИУ9-22	04.05
36	<p>$\text{Stack}\langle T \rangle$ – стек с элементами типа T, имеющий в дополнение к обычным стековым операциям операцию переворачивания стека. Все операции должны работать за константное время за исключением случаев, когда требуется увеличить размер памяти, выделенной для хранения элементов.</p> <p>Для представления стека нужно использовать естественным образом модифицированный кольцевой буфер.</p> <p>В $\text{Stack}\langle \text{string} \rangle$ дополнительно должна быть реализована операция, сообщающая о наличии в стеке пустой строки. Эта операция должна работать за константное время.</p>	Пичугин	ИУ9-22	04.05

Таблица 10: Варианты классов

№	Описание	Студент	Группа	Дата
37	<p>$\text{IntVector}\langle L, H, N \rangle$ – целочисленный вектор размера N с элементами из диапазона от L до H, имеющий следующие операции:</p> <ol style="list-style-type: none"> 1. сложение с другим вектором типа $\text{IntVector}\langle L2, H2, N \rangle$, в результате которого формируется новый вектор типа $\text{IntVector}\langle L+L2, H+H2, N \rangle$; 2. скалярное умножение на вектор типа $\text{IntVector}\langle L2, H2, N \rangle$. <p>Если размер диапазона не превышает 256, для представления вектора должен использоваться массив char'ов.</p>	Прийма	ИУ9-22	04.05
38	<p>$\text{Graph}\langle T \rangle$ – простой граф, рёбра которого имеют атрибуты типа T. Подразумевается, что вершины графа пронумерованы и обращение к ним осуществляется по их номерам. Граф должен быть реализован через матрицу смежности. Граф должен иметь следующие операции:</p> <ol style="list-style-type: none"> 1. определение, смежны ли две вершины; 2. добавление вершины; 3. добавление ребра; 4. получение атрибута ребра, соединяющего две вершины. <p>$\text{Graph}\langle V, \text{int} \rangle$ должен дополнительно иметь операцию построения минимального остовного дерева.</p>	Ростецкий	ИУ9-22	04.05
39	<p>$\text{Tree}\langle K, V, N \rangle$ – двоичное дерево поиска, отображающее ключи типа K в значения типа V, с максимальным размером N. Размер N может быть не задан ($N = 0$), и тогда память для новых вершин должна выделяться динамически. Если же размер N задан, то вершины должны размещаться в массиве размера N, хранящемся в объекте дерева.</p> <p>Для дерева должна быть перегружена операция индексации (по ключу), которая в случае отсутствия искомого ключа в дереве добавляет новую вершину.</p>	Рыбаков	ИУ9-22	04.05

Таблица 11: Варианты классов

№	Описание	Студент	Группа	Дата
40	<p>IntQueue<L, H> – очередь целых чисел из диапазона от L до H, имеющая стандартный для очереди набор операций и реализованная через кольцевой буфер.</p> <p>Если размер диапазона не превышает 256, для представления очереди должен использоваться массив char'ов.</p>	Спиридонова	ИУ9-22	04.05
41	<p>Queue<T, N> – очередь с элементами типа T и максимальным размером N, имеющая обычные для очереди операции и реализованная через массив размера N.</p> <p>В Queue<bool, N> каждый элемент должен быть представлен одним битом.</p>	Актюрк	ИУ9-23	04.05
42	<p>Graph<int N, bool Acyclic> – простой неориентированный граф, у которого N вершин и который может быть ациклическим или не ациклическим. В графе должны быть реализованы операции: добавление ребра (принимает номера вершин, которые надо соединить, отказывается формировать циклы в случае ациклического графа) и проверка смежности двух вершин. Для недопущения формирования циклов можно использовать идею из алгоритма Крускала.</p>	Артеменко	ИУ9-23	04.05
43	<p>MaterialPoints<T, M> – система материальных точек в M-мерном пространстве. Координаты и массы точек заданы числами типа T. Операции: вычисление центра масс; добавление новой точки.</p> <p>В случае, если $M = 2$, в классе должна быть дополнительная операция, вычисляющая минимальную площадь прямоугольника, содержащего все точки системы. Стороны прямоугольника параллельны осям координат.</p>	Бакланова	ИУ9-23	04.05
44	<p>BitmapSet<T, N> – множество натуральных чисел из интервала $[0, N)$, реализованное как битовая маска на базе массива целочисленного типа T. Множество должно поддерживать операцию проверки принадлежности числа, а также операции добавления и удаления числа.</p> <p>Специализированная версия BitmapSet<long, 64> должна использовать для хранения множества не массив, а единственное значение типа long.</p>	Богданов	ИУ9-23	04.05

Таблица 12: Варианты классов

№	Описание	Студент	Группа	Дата
45	<p>IntSet<L, H> – множество целых чисел из диапазона от L до H, реализованное через отсортированный массив и имеющее следующие операции:</p> <ol style="list-style-type: none"> 1. определение принадлежности числа множеству (реализуется путём поиска числа в массиве делением пополам); 2. добавление числа в множество; 3. удаление числа из множества. <p>В случае, если размер диапазона не превышает 64, для представления множества должно использоваться значение типа unsigned long, в котором каждому числу соответствует один бит.</p>	Бокарев	ИУ9-23	04.05
46	<p>Seq<T, N> – неизменяемая упорядоченная последовательность длины N элементов типа T, имеющая операцию, возвращающую значение i-того элемента, и операцию поиска элемента делением пополам.</p> <p>В классе Seq<T, N> должен быть реализован статический метод merge, выполняющий слияние двух упорядоченных последовательностей, имеющих длины N и M, в упорядоченную последовательность длины $N + M$, ссылка на которую передаётся ему в качестве параметра. Конструктор класса Seq<T, N> должен принимать в качестве параметра массив, имеющий тип $T[N]$, и выполнять сортировку этого массива слиянием с использованием метода merge.</p> <p>Конструктор специализированной версии класса Seq<T, 1> просто копирует внутрь объекта передаваемый ему массив единичной длины.</p>	Бостанджян	ИУ9-23	04.05
47	<p>Vector<T, N> – вектор размера N с элементами типа T, имеющий операции сложения и скалярного умножения. Вектор должен быть реализован через массив размера N.</p> <p>В Vector<bool, N> каждый компонент должен быть представлен одним битом.</p>	Дурдымурадова	ИУ9-23	04.05

Таблица 13: Варианты классов

№	Описание	Студент	Группа	Дата
48	<p>Path<L, N> – траектория движения точки на плоскости, заданная с помощью n положений точки в моменты времени $0, 1, \dots, n - 1$. Координаты точки задаются целыми числами, лежащими в диапазоне от L до H. Операции: вычисление длины траектории; добавление нового положения точки в конец траектории. Координаты точек должны храниться в виде пар $\langle x - L, y - L \rangle$, причём для представления каждого элемента пары должен использоваться целочисленный тип минимального размера, подходящего для представления числа $H - L + 1$.</p>	Жданов	ИУ9-23	04.05
49	<p>Equality<T> – линейное равенство вида $a_1x_1 + a_2x_2 + \dots + a_nx_n = b$, коэффициенты a_i и b которого заданы значениями типа T. В классе должна быть предусмотрена операция проверки, удовлетворяет ли указанный вектор значений переменных равенству. В случае, если T – булевский тип, то операция умножения должна трактоваться как логическое И, а операция сложения – как ИЛИ.</p>	Жулева	ИУ9-23	04.05
50	<p>Path<T, N> – траектория движения точки на плоскости, заданная положениями точки в моменты времени $0, 1, \dots, N$. Координаты точки задаются числами типа T. Операции: вычисление длины траектории; добавление другой траектории размера N_2, в результате которого формируется новая траектория размера $N + N_2$. В случае, если N – нечётное число, в классе должна быть дополнительная операция, формирующая траекторию, в которую входят положения точки в моменты времени $0, 2, 4, \dots, N$.</p>	Зайцев	ИУ9-23	04.05
51	<p>RangeSet<T, N> – набор из N интервалов вида (a, b), задающий множество чисел типа T. Операции: проверка принадлежности числа множеству; добавление другого набора интервалов размера N_2, в результате которого формируется новый набор размера $N + N_2$. В специализированной версии RangeSet<char, N> должна дополнительно присутствовать операция проверки принадлежности всех символов строки множеству, задаваемому набором интервалов.</p>	Конюхов	ИУ9-23	04.05

Таблица 14: Варианты классов

№	Описание	Студент	Группа	Дата
52	<p>Route<T, N, M> – маршрут коммивояжера, проходящий через N городов. Матрица расстояний между парами городов размера $M \times M$ передаётся конструктору класса в качестве параметра, причём расстояния задаются числами типа T. Операции: вычисление длины маршрута; добавление другого маршрута размера N_2, в результате которого формируется новый маршрут размера $N + N_2$.</p> <p>В случае, если N – нечётное число, в классе должна быть дополнительная операция, формирующая маршрут, в который входят города исходного маршрута с номерами $1, 3, 5, \dots, N$.</p>	Курушин	ИУ9-23	04.05
53	<p>Polygon<T, M> – многоугольник в M-мерном пространстве, заданный координатами вершин. Координаты вершин представлены числами типа T. Операции: вычисление периметра (возвращает double); добавление новой вершины.</p> <p>В случае, если $M = 2$, в классе должна быть дополнительная операция вычисления минимальной площади прямоугольника, содержащего все вершины многоугольника. Стороны прямоугольника параллельны осям координат.</p>	Лысенко	ИУ9-23	04.05
54	<p>Equality<T, N> – линейное равенство вида $a_1x_1 + a_2x_2 + \dots + a_Nx_N = b$, коэффициенты a_i и b которого заданы числами типа T. В классе должна быть предусмотрена операция проверки, удовлетворяет ли указанный вектор значений переменных равенству.</p> <p>Если $N = 1$, то равенство является линейным уравнением от одной переменной, и в классе должна быть дополнительная операция, решающая это уравнение.</p>	Петров	ИУ9-23	04.05
55	<p>Queue<T, N> – очередь с элементами типа T и максимальным размером N, имеющая в дополнение к обычным для очереди операциям операцию переворачивания очереди, работающую за константное время.</p> <p>В Queue<int, N> должна быть дополнительно реализована операция вычисления суммы элементов, работающая за константное время.</p>	Пинчук	ИУ9-23	04.05

Таблица 15: Варианты классов

№	Описание	Студент	Группа	Дата
56	<p>IntPQueue<L, H> – очередь с приоритетом, предназначенная для целых чисел из диапазона от L до H и имеющая стандартный для очереди с приоритетом набор операций.</p> <p>Если размер диапазона не превышает 256, для представления очереди должен использоваться массив char'ов.</p>	Подольный	ИУ9-23	04.05
57	<p>RangeSet<L, H> – набор интервалов вида $[a, b]$, где a и b – такие целые числа, что $L \leq a, b \leq H$.</p> <p>Операции: проверка принадлежности числа хотя бы одному интервалу набора; добавление в набор нового интервала.</p> <p>Каждый интервал должен храниться в виде пары $\langle a, \delta \rangle$, в которой a – нижняя граница, а $\delta = b - a$ – размер интервала. Тип для представления δ должен быть целочисленным типом минимального размера, подходящего для представления числа $H - L + 1$.</p>	Рогов	ИУ9-23	04.05
58	<p>Queue<T, N> – очередь с элементами типа T и максимальным размером N, реализованная через двойной стек и имеющая обычные для очереди операции.</p> <p>Queue<int, N> дополнительно имеет операцию нахождения максимального числа в очереди.</p>	Снегур	ИУ9-23	04.05
59	<p>FixNum<N, M> – число с фиксированной точкой, реализованное на базе массива int'ов размера N и имеющее M бит, отведённых для хранения дробной части. Число должно поддерживать операции: сложение с другим таким же числом (при сложении не поместившийся бит переноса отбрасывается), перевод в число с плавающей точкой.</p> <p>Специализированная версия FixNum<1, M> должна использовать для хранения числа не массив, а единственное значение типа int.</p>	Сырбу	ИУ9-23	04.05
60	<p>Rat<class T, bool Finite> – рациональное число, числитель и знаменатель которого имеют тип T, которое может принимать значения $+\infty$ и $-\infty$, если $Finite = false$. Для рациональных чисел нужно реализовать операцию сравнения (возвращает целое число наподобие функции strcmp языка C) и операцию сложения.</p>	Чигвинцев	ИУ9-23	04.05

Таблица 16: Варианты классов

№	Описание	Студент	Группа	Дата
61	Matrix<M,N> – вещественная матрица размера $M \times N$ с операцией доступа к элементу (возвращает ссылку) и операцией сложения. В случае, если матрица является вектором ($M = 1$ или $N = 1$), для неё должна быть доступна операция вычисления длины вектора.	Чурсина	ИУ9-23	04.05
62	Vandermonde<T, M, N> – матрица Вандермонда размера $M \times N$. Элементы матрицы Вандермонда вычисляются по формуле $V_{i,j} = \alpha_i^{j-1}$ и задаются набором из M значений коэффициентов α_i типа T . Операции: вычисление элемента $V_{i,j}$ матрицы; вычисление определителя (операция доступна, если $M = N$).	Шевляков	ИУ9-23	04.05
63	Number<D,N> – целое беззнаковое число, представленное последовательностью цифр в системе счисления по основанию D , где $1 \leq D < 2^{31}$. Длина последовательности ограничена сверху числом N . Для чисел должна быть реализована операция сложения и операция, возвращающая ссылку на i -тую цифру. В случае, когда $D = 1$, число представлено в так называемой унарной системе счисления, в которой значение числа равно количеству цифр в нём.	Ширяева	ИУ9-23	04.05
64	MaterialPoints<T, N> – система из N материальных точек на плоскости. Координаты и массы точек заданы числами типа T . Операции: вычисление центра масс; добавление другого множества размера N_2 , в результате которого формируется новое множество размера $N + N_2$. В случае, если N – нечётное число, в классе должна быть дополнительная операция, возвращающая медиану системы, т.е. такую точку, что если отсортировать все точки системы по возрастанию массы, она будет располагаться ровно в середине получившейся последовательности.	Яушев	ИУ9-23	04.05