## Задача № 1 «Представление графа».(4 балла)

Дан базовый интерфейс для представления ориентированного графа:

struct IGraph {

virtual ~IGraph() {}

// Добавление ребра от from к to.

virtual void AddEdge(int from, int to) = 0;

virtual int VerticesCount() const = 0;

virtual void GetNextVertices(int vertex, std::vector<int>& vertices) const = 0;

virtual void GetPrevVertices(int vertex, std::vector<int>& vertices) const = 0;

};

Необходимо написать несколько реализаций интерфейса:

* ListGraph, хранящий граф в виде массива списков смежности,
* MatrixGraph, хранящий граф в виде матрицы смежности,
* SetGraph, хранящий граф в виде массива хэш-таблиц,
* ArcGraph, хранящий граф в виде одного массива пар {from, to}.

Также необходимо реализовать конструктор, принимающий const IGraph\*. Такой конструктор должен скопировать переданный граф в создаваемый объект.  
Для каждого класса создавайте отдельные h и cpp файлы.  
Число вершин графа задается в конструкторе каждой реализации.

## Задача № 2. Цикл минимальной длины (2 балла)

Дан невзвешенный неориентированный граф. Найдите цикл минимальной длины.

Ввод: v:кол-во вершин(макс. 50000), n:кол-во ребер(макс. 200000), n пар реберных вершин

Вывод: одно целое число равное длине минимального цикла. Если цикла нет, то вывести -1.

|  |  |
| --- | --- |
| in | out |
| 6  6 0 1 0 2 2 3 2 4 3 5 4 5 | 4 |

## Задача № 3. Количество различных путей (2 балла)

Дан невзвешенный неориентированный граф. В графе может быть несколько кратчайших путей между какими-то вершинами. Найдите количество различных кратчайших путей между заданными вершинами. Требуемая сложность O(V+E).

Ввод: v:кол-во вершин(макс. 50000), n:кол-во ребер(макс. 200000), n пар реберных вершин, пара вершин v, w для запроса.

Вывод:количество кратчайших путей от v к w

|  |  |
| --- | --- |
| in | out |
| 4 5 0 1 0 2 1 2 1 3 2 3  0 3 | 2 |

## Задача № 4. Двудольный граф (2 балла)

Дан невзвешенный неориентированный граф. Определить, является ли он двудольным. Требуемая сложность O(V+E).

Ввод: v:кол-во вершин(макс. 50000), n:кол-во ребер(макс. 200000), n пар реберных вершин.

Вывод: YES если граф является двудольным, NO - если не является.

|  |  |
| --- | --- |
| in | out |
| 3  3  0 1  1 2  0 2 | NO |
| 5  6  0 1  0 2  0 3  1 4  2 4  3 4 | YES |

## Задача № 5. Планарный граф (5 баллов)

Дан невзвешенный неориентированный граф. Определить, является ли он планарным.

Ввод: v:кол-во вершин(макс. 1000), n:кол-во ребер(макс. 3000), n пар реберных вершин.

Вывод: YES если граф является планарным, NO - если не является.

|  |  |
| --- | --- |
| in | out |
| 3  3  0 1  1 2  0 2 | YES |
| 6  9  0 3  0 4  0 5  1 3  1 4  1 5  2 3  2 4  2 5 | NO |

## Задача № 6. Дополнение до сильносвязного (5 баллов)

Дан ориентированный граф. Определите, какое минимальное количество ребер необходимо добавить, чтобы граф стал сильносвязным. В графе возможны петли.