

Лабораторная работа № 3

по курсу “Языки программирования и методы программирования” (информатика, 3 семестр)

Техническое задание

1. Постановка задачи

Написать программу на C++, реализующую алгоритмы (поиска) на графах.
Написать краткое техническое задание (ТЗ). Выполнить реализацию.
Написать для нее тесты.

2. Функциональные требования

Ориентированный граф должен позволять хранить в своих ребрах любые численные типы, то есть быть шаблонным.

Методы класса:

Метод:	Назначение:	Сигнатура:
PrintMatrix	Печатает матрицу смежности графа	void PrintMatrix()
Dijkstra	Ищет все кратчайшие пути от заданной вершины методом Дейкстры	Sequence<int>* Dijkstra(int start, int end)
result_path	Выбирает путь от заданной вершины до нужной пользователю, используя информацию из предыдущего метода	Sequence<int>* result_path(Sequence<int>* ex_path, Sequence<T>* short_path, int start, int end)
Length	Возвращает длину пути	T Length(Sequence<int>* path)
PrintPath	Печатает вершины в порядке их прохождения	void PrintPath(Sequence<int>* path)

Если запрашивается путь из начальной вершины в нее же, то он считается равным нулю.

Если пути из начальной вершины в конечную не существует, он считается равным бесконечности.

Результат является одним кратчайшим путем, а не набором всех возможных путей из начальной вершины.

3. Требования к структурам данных

Ориентированный граф должен быть основан на матрице смежности, которая в свою очередь базируется на виртуальном классе Sequence, пользующийся реализацией методов ArraySequence. То есть матрица представляется идущими друг за другом строками в массиве.

Основные методы класса Sequence:

Метод:	Назначение:	Сигнатура:
GetSize	Возвращает размер подпоследовательности	int GetSize() const
Set	Вставляет элемент по индексу	void Set(int index, T item)
Get	Возвращает элемент по индексу	T Get(const int i) const
Delete_	Удаляет элемент по индексу	void Delete_(const int index)
Append	Вставляет элемент в конец	void Append(T value)
Prepend	Вставляет элемент в начало	void Prepend(T value)

4. Требования к интерфейсу

Должен быть реализован консольный интерфейс.

Интерфейс должен предоставлять пользователю выбор между задачами, поддерживать как автоматический, так и ручной ввод данных, давать возможность пользователю увидеть результат и промежуточные значения, а также время выполнения алгоритмов.

5. Требования к входным и выходным данным

Данные должны являться целыми числами.

6. Требования к тестированию

Все основные методы должны быть покрыты тестами.