

Требования к представлению данных в программах:

Граф (задания 1–7, 9–11)

Граф хранится в классе `Graph` и содержит следующие открытые функции:

конструктор класса, принимающий путь к файлу и параметр, в котором указывается тип файла;

`weight(Vertex vi, Vertex vj)` – весовая функция, принимает номера вершин, возвращает вес ребра, связывающего их;

`is_edge(Vertex vi, Vertex vj)` – функция, принимающая номера вершин и возвращает `True`, если в графе есть соответствующее ребро/дуга, `False`, если ребра нет;

`adjacency_matrix()` – функция, возвращающая матрицу смежности графа/орграфа;

`adjacency_list(Vertex v)` – функция, возвращающая список вершин, смежных вершине `v`;

`list_of_edges()` – функция, возвращающая список всех рёбер графа;

`list_of_edges(Vertex v)` – функция, возвращающая список всех рёбер графа, инцидентных вершине `v` / исходящих из вершины `v`.

`is_directed()` – функция возвращающая `True`, если граф ориентированный, `False`, если граф простой;

Карта (задание 8)

Для обращения к карте используется вспомогательный класс `Cell`, в котором хранятся координаты.

Карта хранится в классе `Map` и содержит следующие открытые функции:

конструктор класса, принимающий путь к файлу;

Индексатор, возвращающий значение высоты в точке;

`neighbors(Cell c)` – функция, возвращающая список клеток, соседних клетке `c`.

Требования к входным файлам:

1. Граф, заданный списком рёбер. Каждое ребро хранится отдельной строкой. В строке три числа, разделённых пробелом: исходящая вершина, входящая вершина, вес ребра. Номера вершин представляют собой натуральные числа с нулём. Размерность графа определяется автоматически (от 0 вершины до вершины с наибольшим номером).
2. Граф, заданный списками смежности. Номер строки представляет собой номер вершины графа, в строке через пробел перечислены номера смежных вершин.
3. Граф, заданный матрицей смежности. Значения в строках разделены пробелами.
4. Карта, представляющая собой прямоугольную матрицу, где i -я строка и j -й столбец задают «клетку» на карте. Значение представляет собой высоту местности в данной точке. Переход возможен только между соседними «клетками» по вертикали или горизонтали. Расстояние между клетками выражается следующей функцией: $h(ij, kl) = abs(k - i) + abs(l - j) + abs(a_{kl} - a_{ij})$, где ij – координаты текущей «клетки», kl – координаты соседней клетки, a_{ij} – значение в текущей «клетке», обозначающее высоту в этой точке, a_{kl} – значение высоты в «клетке» с координатами kl . Эвристики задаются следующими функциями (sp – координаты конечной точки маршрута):
 - a. Манхэттенское расстояние $d(ij, sp) = abs(s - i) + abs(p - j)$.
 - b. Расстояние Чебышева $d(ij, sp) = \max(abs(s - i), abs(p - j))$.
 - c. Евклидово расстояние $d(ij, sp) = \sqrt{(s - i)^2 + (p - j)^2}$.
 - d. Эвристика отсутствует (алгоритм Дейкстры) $d(ij, sp) = 0$.

1. Программа, рассчитывающая следующие характеристики графа/орграфа: вектор степеней вершин, матрицу расстояний, диаметр, радиус, множество центральных вершин (для графа), множество периферийных вершин (для графа). Расчёт производится алгоритмом Флойда-Уоршелла. Входные данные для работы программы задаются следующими ключами с параметрами:

`-e "edges_list_file_path"`

`-m "adjacency_matrix_file_path"`

`-l "adjacency_list_file_path"`

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

2. Программа, определяющая связность. Для графа – связность, а также количество и состав компонент связности. Для орграфа – сильную, слабую связность, или несвязность. А также количество и состав компонент связности и сильной связности. Для определения используется поиск в ширину.

Входные данные для работы программы задаются следующими ключами с параметрами:

`-e "edges_list_file_path"`

`-m "adjacency_matrix_file_path"`

`-l "adjacency_list_file_path"`

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

3. Программа, находящая мосты и шарниры в графе. Для орграфа находятся мосты и шарниры в соотнесённом графе.

Входные данные для работы программы задаются следующими ключами с параметрами:

`-e "edges_list_file_path"`

`-m "adjacency_matrix_file_path"`

`-l "adjacency_list_file_path"`

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

4. Программа, находящая остовное дерево графа. Для орграфа находится остовное дерево соотнесённого графа. Результатом является список рёбер графа, входящих в остовное дерево и суммарный вес дерева.

Входные данные для работы программы задаются следующими ключами с параметрами:

-e "edges_list_file_path"
-m "adjacency_matrix_file_path"
-l "adjacency_list_file_path"

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Алгоритм для расчёта задаётся следующими ключами:

-k – алгоритм Крускала
-p – алгоритм Прима
-b – алгоритм Борувки
-s – расчёт производится тремя алгоритмами поочерёдно

Результатом в первых трёх случаях является список рёбер, составляющих остовное дерево. В четвёртом случае дополнительно выводится время работы алгоритма в мс. Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

-o "output_file_path"

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа -h. При этом, если указаны остальные ключи, они игнорируются.

5. Программа, находящая геодезическую цепь между двумя вершинами в графе. Поиск производится алгоритмом Дейкстры. Результатом работы является маршрут между вершинами, заданный списком рёбер, и длина маршрута.

Входные данные для работы программы задаются следующими ключами с параметрами:

-e "edges_list_file_path"
-m "adjacency_matrix_file_path"

-l "adjacency_list_file_path"

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Начальная вершина задаётся следующим ключом:

-n begin_vertex_number

Данный ключ является обязательным.

Конечная вершина задаётся следующим ключом:

-d end_vertex_number

Данный ключ является обязательным.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

-o "output_file_path"

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа -h. При этом, если указаны остальные ключи, они игнорируются.

6. Программа, рассчитывающая расстояние от указанной вершины до всех остальных вершин в графе. Результатом работы является перечисление пар вершин, и соответствующих расстояний между ними.
Входные данные для работы программы задаются следующими ключами с параметрами:

-e "edges_list_file_path"

-m "adjacency_matrix_file_path"

-l "adjacency_list_file_path"

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Алгоритм для расчёта задаётся следующими ключами:

- d – алгоритм Дейкстры
- b – алгоритм Беллмана-Форда-Мура
- t – алгоритм Левита

Если граф/орграф не удовлетворяет требованиям алгоритма к входным данным, выдать пользователю соответствующее предупреждение.

Начальная вершина задаётся следующим ключом:

- n begin_vertex_number

Данный ключ является обязательным.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

- o "output_file_path"

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа -h. При этом, если указаны остальные ключи, они игнорируются.

7. Программа, рассчитывающая расстояние между всеми парами вершин в графе. Поиск производится алгоритмом Джонсона. Результатом работы является перечисление пар вершин, и соответствующих расстояний между ними.

Входные данные для работы программы задаются следующими ключами с параметрами:

- e "edges_list_file_path"
- m "adjacency_matrix_file_path"
- l "adjacency_list_file_path"

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

8. Программа, рассчитывающая расстояние между двумя точками на карте. Поиск производится алгоритмом A^* . Результатом работы является маршрут между вершинами, заданный списком пройденных точек. Входной файл для работы программы задаётся следующим ключом:

`-m "map_file_path"`

Начальная вершина задаётся следующим ключом с параметрами:

`-n begin_vertex_x begin_vertex_y`

Данный ключ является обязательным.

Конечная вершина задаётся следующим ключом:

`-d end_vertex_x end_vertex_y`

Данный ключ является обязательным.

Алгоритм запускается поочерёдно со всеми эвристиками. Для каждой эвристики выводится найденный маршрут, его длина и процент просмотренных алгоритмом клеток от общего их числа.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается

при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

9. Программа, находящая гамильтонов путь в графе. Результатом работы является маршрут, заданный списком рёбер.
Входные данные для работы программы задаются следующими ключами с параметрами:

`-e "edges_list_file_path"`
`-m "adjacency_matrix_file_path"`
`-l "adjacency_list_file_path"`

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Алгоритм для расчёта задаётся следующими ключами:

`-b` – метод ветвей и границ
`-a` – метод муравьиной колонии

Если граф/орграф не удовлетворяет требованиям алгоритма к входным данным, выдать пользователю соответствующее предупреждение.
Начальная вершина задаётся следующим ключом:

`-n begin_vertex_number`

Данный ключ является обязательным.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

10. Программа, рассчитывающая максимальный поток в сети. Расчёт выполняется алгоритмом Форда-Фалкерсона. Источник и сток определяются автоматически. Результатом работы является список рёбер с указанием величины потока.

Входные данные для работы программы задаются следующими ключами с параметрами:

```
-e "edges_list_file_path"

-m "adjacency_matrix_file_path"

-l "adjacency_list_file_path"
```

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Если граф/орграф не удовлетворяет требованиям алгоритма к входным данным, выдать пользователю соответствующее предупреждение.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

```
-o "output_file_path"
```

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.

11. Программа, находящая максимальные паросочетания. Перед этим определяется, является ли граф двудольным. Результатом является список рёбер (паросочетаний). Для орграфа находятся паросочетания соотнесённого графа.

Входные данные для работы программы задаются следующими ключами с параметрами:

```
-e "edges_list_file_path"

-m "adjacency_matrix_file_path"
```

`-l "adjacency_list_file_path"`

Одновременно может указываться только один из этих ключей. Если указано более одного – выдать сообщение об ошибке.

Если граф/орграф не удовлетворяет требованиям алгоритма к входным данным, выдать пользователю соответствующее предупреждение.

Результаты работы выводятся на экран, либо в файл при указании следующего ключа:

`-o "output_file_path"`

Также должна быть доступна справка, в которой указывается: автор работы, группа, список ключей с кратким описанием. Справка вызывается при помощи ключа `-h`. При этом, если указаны остальные ключи, они игнорируются.