руемого текста, то такой шифр теоретически является абсолютно стойким, т.е. его нельзя вскрыть при помощи статистической обработки зашифрованного текста, а можно раскрыть только прямым перебором (brootforce). Криптостойкость в этом случае определяется размером ключа. Это, однако, не означает, что дешифрование такого текста вообще невозможно: при наличии некоторой дополнительной информации исходный текст может быть частично или полностью восстановлен даже при использовании бесконечной гаммы.

Кроме того, для использования ключа вначале необходимо каким-либо надежным способом доставить его обеим сторонам, обменивающимся сообщениями. Это приводит к возникновению проблемы распределения ключей, сложность решения которой возрастает с увеличением длины ключа и количества абонентов в сети передачи сообщений.

В качестве гаммы может быть использована любая последовательность случайных символов, например, последовательность цифр числа π, числа e и т.п. При шифровании с помощью ЭВМ последовательность гаммы формируется с помощью ***генератора псевдослучайных чисел*** (ГПСЧ). В настоящее время разработано несколько алгоритмов работы таких генераторов, которые обеспечивают удовлетворительные характеристики гаммы.

ЛИТЕРАТУРА

1. Методы гаммирования [Электронный ресурс] – Режим доступа: https://www.intuit.ru/studies/courses/691/547/lecture/12373?page=4. – Дата обращения: 05.04.2019.

2. Шифрование методом гаммирования [Электронный ресурс] – Режим доступа: https://studfiles.net/preview/5470123/page:9/ – Дата обращения: 08.04.2019.

УДК 519.176

Студ. Р. Ю. Злобин
Науч. рук. доц. И. К. Асмыкович
(кафедра высшей математики, БГТУ)

НЕКОТОРЫЕ ПРИМЕНЕНИЯ ТЕОРИИ ГРАФОВ

Графы и теория графов представляют собой один из наиболее важных и интересных, но в тоже время один из сложных разделов в математике и информатике.

Многие структуры, представляющие практический интерес в логике, информатике, математике и других науках, могут быть представлены графами.

Теория графов находит применение, например, в геоинформационных системах (ГИС), химии, экономике, логистике, схемотехнике. Применение различных вычислений, производимых на графах, позволяет, например, найти кратчайший объездной путь или спланировать оптимальный маршрут.

Цель работы: описать некоторые задачи теории графов и алгоритмы их решения.

Граф G – это упорядоченная пара $G := (V, E)$, где V – это непустое множество вершин или узлов, а E – множество пар вершин, называемых рёбрами [1].

Вершины и рёбра графа называются также элементами графа, число вершин в графе — порядком, число рёбер — размером графа.

Граф может быть ориентированным или неориентированным. В ориентированном графе, связи являются направленными (то есть пары в множестве E являются упорядоченными, например, пары (2,5) и (5,2) это разные связи). В неориентированном графе, связи ненаправленные, и поэтому если существует связь (2,5), то значит, что и связь (5,2) существует.

При изображении графов на рисунках чаще всего используется следующая система обозначений: вершины графа изображаются точками или, при конкретизации смысла вершины, прямоугольниками, овалами и др., где внутри фигуры раскрывается смысл вершины (графы блок-схем алгоритмов). Если между вершинами существует ребро, то соответствующие точки (фигуры) соединяются линией или дугой. В случае ориентированного графа дуги заменяют стрелками, они явно указывают направленность ребра. Иногда рядом с ребром размещают поясняющие надписи, раскрывающие смысл ребра, например, в графах переходов конечных автоматов.

При создании программ для работы с графами большую значимость имеет метод их представления в памяти компьютера. Есть два основных метода представления графа: в виде списком смежности и в виде матрицы смежности.

Матрица смежности представляет собой квадратную матрицу A размером $|V| \times |V|$, где V – это число вершин в графе. Каждый элемент матрицы A несёт в себе информацию о рёбрах графа, например, наличие или отсутствие ребра из вершины i в j (обычно обозначается 1 или 0).

При списках смежности используется массив A, содержащий $|V|$ списков, где V – это число вершин в графе. Список $A[i]$ содержит такие вершины u, что между i и u есть ребро.

Каждый из данных способов имеет свои преимущества и недостатки. Выбор каждого из них зависит от типа задачи, вида исходных данных, предполагаемого решения и т.д.

В интернете, а также в специализированной литературе существует большое множество задач, решение которых представляется в виде графа.

Одной из таких задач является поиск кратчайших расстояний от одной вершины ко всем остальным.

Имеется орграф. Необходимо найти кратчайшие расстояния от вершины 1 до остальных, построить дерево кратчайших путей.

Для решения задачи воспользуемся алгоритмом Дейкстры [3]. Он заключается в том, что вершинам графа присваиваются временные метки d_i, которые меняются по определённым правилам. У всех меток вершин, кроме стартовой, принимают значение $\propto$. У стартовой метки значение 0.

Затем, до тех пор, пока минимальное значение меток не равно $\propto$ или не прошло n итераций (n – число вершин графа), производим релаксации из вершины с минимальной меткой, т.е. если i – индекс вершины с минимальной меткой, а j – индекс сопряжённой вершины с вершиной i, то $d_j = \min(d_j, d_i + a_{i,j})$, где $a_{i,j}$– вес ребра между вершинами i и j.

Для построения дерева кратчайших путей, запоминаем для каждой вершины индекс вершины p_i, из которой проводилась её последняя релаксация. После в исходном графе для каждой вершины оставляем только то ребро, которые идёт в неё из p_i.

Ещё одной типовой задачей является, получение дерева минимального веса.

Дан взвешенный связный неориентированный граф. Необходимо удалить рёбра графа, так что бы получилось дерево, причём сумма весов его рёбер должна быть минимальна. Называется минимальным остовом.

Обычно данную задачу решают с помощью алгоритмов Прима и Краскала.

Алгоритм Прима. Искомое дерево строится постепенно, добавляют в него рёбра по одному. Изначально остов состоит из одной вершины (её можно выбрать произвольно). Затем выбирается ребро минимального веса, исходящее из этой вершины, и добавляется в остов. После этого остов содержит уже две вершины, и теперь ищут и добавляют ребро минимального веса, имеющее один конец в одной из двух выбранных вершин, а другой — наоборот, во всех остальных, кроме этих двух. И так далее, т.е. всякий раз ищется минимальное по

весу ребро, один конец которого — уже взятая в остов вершина, а другой конец — ещё не взятая, и это ребро добавляется в остов (если таких рёбер несколько, можно взять любое). Этот процесс повторяется до тех пор, пока остов не станет содержать все вершины (или, что то же самое, $n - 1$ ребро).

В итоге будет построен остов, являющийся минимальным. Если граф был изначально не связен, то остов найден не будет.

Алгоритм Краскала. Изначально помещают каждую вершину в своё дерево, а затем постепенно объединяют эти деревья, объединяя на каждой итерации два некоторых дерева ребром. Перед началом выполнения алгоритма, все рёбра сортируются по весу (в порядке возрастания). Затем начинается процесс объединения: перебираются все рёбра от первого до последнего (в порядке сортировки), и если у текущего ребра его концы принадлежат разным поддеревьям, то эти поддеревья объединяются, а ребро добавляется к ответу.

В итоге все вершины окажутся в одном поддереве, которое и образует минимальный остов.

Графы являются универсальным средством, которое помогает нам как при решении задач в различных практических сферах. Умение оперировать ими, необходимых навык востребованного специалиста.

ЛИТЕРАТУРА

1. Граф (математика) Wikipedia [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0). – Дата доступа: 29.03.2019.

2. Оре О. Теория графов. — 2-е изд. — М.: Наука, 1980. — С. 336.

3. Иллюстративное введение в теорию графов и её применение Proglib [Электронный ресурс]. – Режим доступа: https://proglib.io/p/graph-theory/– Дата доступа: 29.03.2019.