FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION

OF HIGHER EDUCATION

ITMO UNIVERSITY

Report

on the practical task No. 8

“Practical analysis of advanced algorithms”

Performed by

Pakulev Aleksandr

J4133c

Accepted by

Dr Petr Chunaev

St. Petersburg

2021

Goal
Practical analysis of advanced algorithms

Task

1. Choose two algorithms (interesting to you and not considered in the course) from the above-mentioned book sections.
2. Analyse the chosen algorithms in terms of time and space complexity, design technique used, etc. Implement the algorithms and produce several experiments. Analyse the results.

I choose

1. Huffman coding which is greedy algorithm
2. Prim algorithm which is algorithm on graph and greedy too

Brief theoretical part:

Huffman Coding is a technique of compressing data to reduce its size without losing any of the details. It was first developed by David Huffman. Huffman Coding is generally useful to compress the data in which there are frequently occurring characters.

1. At the first step we Calculate the frequency of each character in the string.
2. Then, Sort the characters in increasing order of the frequency. These are stored in a priority queue
3. Make each unique character as a leaf node.
4. Create an empty node z. Assign the minimum frequency to the left child of z and assign the second minimum frequency to the right child of z. Set the value of the z as the sum of the above two minimum frequencies.
5. Remove these two minimum frequencies from Q and add the sum into the list of frequencies (* denote the internal nodes in the figure above).
6. Insert node z into the tree.
7. Repeat steps above while Q not empty.

Huffman Coding is a greedy algorithm because on each step we take the best result from queue. Time complexity O(n * log(n)), where n is input text, Space complexity is O(n).

Prim algorithm. A weighted undirected graph G with n vertices and m edges is given.

You need to find a subtree of this graph that connects all its vertices, and at the same time has the lowest possible weight (i.e., the sum of the weights of the edges). A

subtree is a set of edges that connect all vertices, and from any vertex you can get to any other one in exactly one simple way. Such a subtree is called a minimal spanning tree, or simply a minimal spanning tree.

Any spanning tree will necessarily contain n-1 edges.

Algorithm too simple:

At each step we add to spanning tree edge with minimal weight from not chosen set. Then at each step we found minimum edge where one node is in initial set on nodes and second node in all except this nodes.

Time complexity O(n^2 + m), space complexity O(n+m)

Result:

1) Huffman algorithm:

Here simple line: BCAADDDCCACACAC

And Huffman tree:

Char | Huffman code

'C'| 0

'A' | 11

'D' | 101

'B' | 100

In line above C insert 6 times

A – 5 times

D – 3 times

B – only once

For line: AAAAAAB

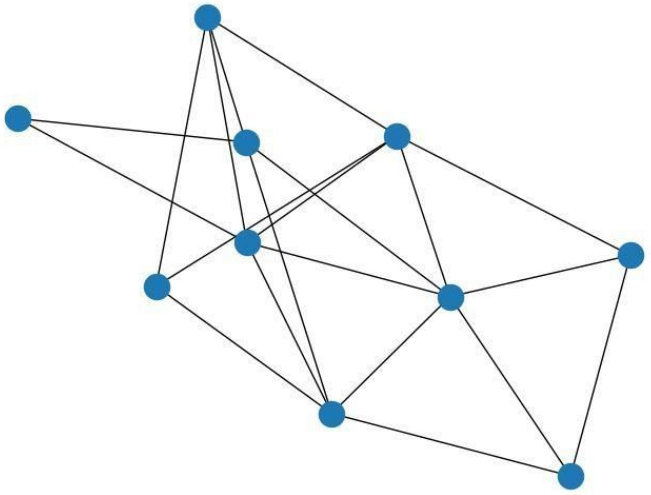
Huffman tree is:

Char | Huffman code

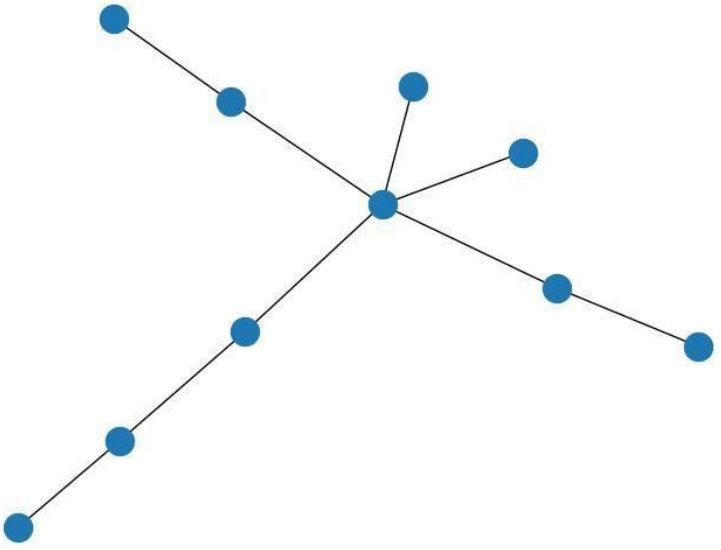
'A'| 1

'B'| 0

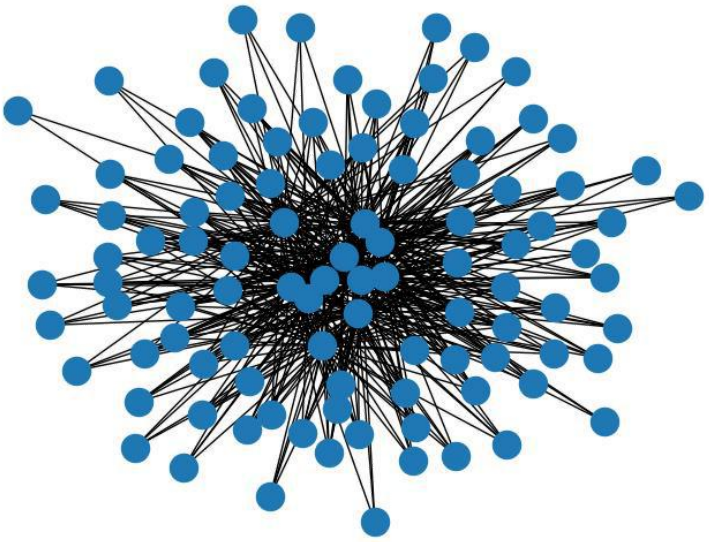
Now consider Prim algorithm:



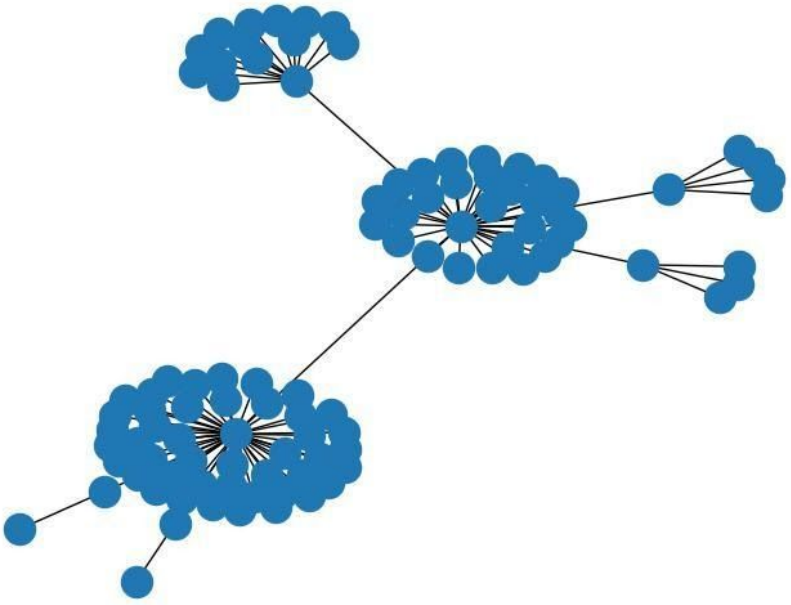
Graph with 10 nodes



Min spanning tree



Graph with 100 nodes



Min spanning tree

Conclusion:

In this task I implemented graph data strucute. And algorithm for determine minimum spanning tree. Additionally I implement Huffman algorithm, which is one of optimal algorithm for short encoding.