Министерство образования Республики Беларусь

Учреждение образования

“Брестский государственный технический университет”

Кафедра интеллектуально-информационных технологий

Лабораторная работа №6

По дисциплине «Языковые процессы интеллектуальных систем»

Выполнил:

студент 4 курса

группы ИИ-22

Копанчук Е. Р.

Проверил:

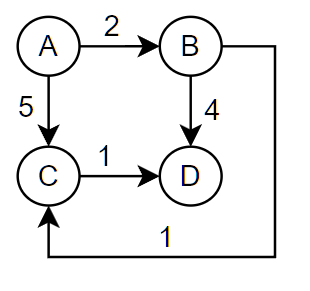
Монтик Н. С.

Брест-2024

**Цель работы:** сформировать представление о методе и алгоритме поиска в глубину на графе и его применении при решении задач.

**Ход работы:**

**Задание:** Напишите программу, которая использует поиск в глубину для нахождения кратчайшего пути в графе с неотрицательными весами рёбер. Граф представлен списком смежности, и вы должны использовать динамические структуры данных.



**Код программы:**

type Graph = {

[key: string]: { node: string, weight: number }[];

};

function dfs(

graph: Graph,

start: string,

end: string,

visited: Set<string>,

path: string[],

currentWeight: number,

bestPath: string[],

bestWeight: { value: number }

) {

if (start === end) {

if (currentWeight < bestWeight.value) {

bestWeight.value = currentWeight;

bestPath.length = 0;

bestPath.push(...path);

}

return;

}

visited.add(start);

for (const neighbor of graph[start]) {

if (!visited.has(neighbor.node)) {

path.push(neighbor.node);

dfs(graph, neighbor.node, end, visited, path, currentWeight + neighbor.weight, bestPath, bestWeight);

path.pop();

}

}

visited.delete(start);

}

function findShortestPath(graph: Graph, start: string, end: string): { path: string[], weight: number } {

const visited = new Set<string>();

const bestPath: string[] = [];

const bestWeight = { value: Infinity };

dfs(graph, start, end, visited, [start], 0, bestPath, bestWeight);

return { path: bestPath, weight: bestWeight.value };

}

const graph: Graph = {

'A': [{ node: 'B', weight: 2 }, { node: 'C', weight: 5 }],

'B': [{ node: 'C', weight: 1 }, { node: 'D', weight: 4 }],

'C': [{ node: 'D', weight: 1 }],

'D': []

};

const result = findShortestPath(graph, 'A', 'D');

console.log('Кратчайший путь:', result.path);

console.log('Вес пути:', result.weight);

**Результат:**

