|  |  |
| --- | --- |
| **NAME:** | Akshaya Karande |
| **UID:** | 2021300056 |
| **SUBJECT** | Design and Analysis of Algorithms |
| **EXPERIMENT NO:** | 10 |
| **AIM:** | To implement the vertex cover problem using approximation algorithm. |
| **AlGORITHM:** | Vertex Cover Problem – Approximation Algorithm  Approx-Vertex-Cover (G = (V, E))  {  C = empty-set;  E'= E;  While E' is not empty do  {  Let (u, v) be any edge in E': (\*)  Add u and v to C;  Remove from E' all edges incident to  u or v;  }  Return C;  } |
| **CODE:** | **Source Code:**  **#include <bits/stdc++.h>**  **using namespace std;**  **struct Edge {**  **char u;**  **char v;**  **Edge() {**  **u = 0;**  **v = 0;**  **}**  **Edge(char u, char v) {**  **this->u = u;**  **this->v = v;**  **}**  **};**  **set<char> vertexCover(vector<Edge>& edges) {**  **set<char> cover;**  **vector<Edge> edges\_copy = edges;**  **srand(time(NULL));**  **while (!edges\_copy.empty()) {**  **int i = rand() % edges\_copy.size();**  **Edge e = edges\_copy[i];**  **edges\_copy.erase(edges\_copy.begin() + i);**  **cover.insert(e.u);**  **cover.insert(e.v);**  **cout << "Adding edge " << e.u << " " << e.v << "\n";**  **for (int i = 0; i < edges\_copy.size(); i++) {**  **if (edges\_copy[i].u == e.u || edges\_copy[i].v == e.u ||**  **edges\_copy[i].u == e.v || edges\_copy[i].v == e.v) {**  **cout << "Removing edge " << edges\_copy[i].u << " "**  **<< edges\_copy[i].v << "\n";**  **edges\_copy.erase(edges\_copy.begin() + i);**  **i--;**  **}**  **}**  **}**  **return cover;**  **}**  **int main() {**  **cout << "Enter the number of edges then enter each edge in the format \"u "**  **"v\" where u and v are vertices of the edge.\n";**  **int n;**  **cin >> n;**  **vector<Edge> edges(n);**  **for (int i = 0; i < n; i++) {**  **char u, v;**  **cin >> u >> v;**  **edges[i] = Edge(u, v);**  **}**  **cout << "\nApproximate Vertex Cover Algorithm\n";**  **set<char> cover = vertexCover(edges);**  **cout << "\nVertex Cover: { ";**  **for (char v : cover) {**  **cout << v << ", ";**  **}**  **cout << "}\n" << endl;**  **return 0;**  **}** |
| **Output** |  |
| **CONCLUSION:** | In conclusion, the experiment using approximation algorithm to solve the Vertex Cover problem was successful in finding a solution that is close to the optimal solution. Although the algorithm may not guarantee an optimal solution, it provides a trade-off between computation time and solution quality. Through the experiment, we observed that the size of the vertex cover produced by the algorithm was within a factor of two of the optimal solution. Overall, the experiment provides insights into the effectiveness of approximation algorithms and their potential applications in solving complex computational problems.  Complexity Analysis:  Time Complexity: O(n + m) // n = number of nodes, m = number of edges  Space Complexity: O(n) |