Algorithms
Vertex Cover Handout

1 Introduction

You are in charge of security at the Museum of Algorithms. The museum's corridors are filled with priceless objects that need to be constantly watched, you having taken an algorithms class decide to represent this as a graph and see that it is the Minimum Vertex Cover Problem. The Graph has m edges with n verticies between them. Your goal is to find the minimum number of verticies, $g \leq n$ you need to cover so that all m edges are being watched by at least one edge

2 Example

Consider the graph instance below:
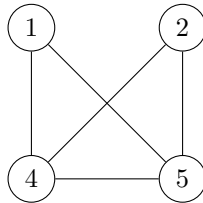
Here we find that the minimum vertex cover is 2, where we choose nodes 4 and 5. Because when we choose these two nodes, the set of edges connected them to is equal to all the edges in the graph. In other words, if this graph represents the museam and we we station our guards at these two intersections, 4 and 5, all the hallways are being watched.

3 Solution Methods

Two ideas (described in English) for algorithms are described next (in English). For pseudocode, see the next section.

3.1 Exhaustive Approach

Create a power set of the set of all vertices. You then iterate through the power set and if a set is covers all hallways and it has fewer guards than the current minimum, save the subset. Continue until you have fully iterated through the power set.

3.2 Greedy Heuristic Approach

You have a list of all possible guard placements (intersections of hallways) and the number of hallways they watch. You then put the guard at the intersection watching the most hallways, if there is a tie you choose the intersection that has the alphabetically lower label. You then update the list of placements and the respective hallway count subtracting those that are now watched by the guards. Repeat this until all hallways are being watched.

4 Pseudocode for exhaustive and heuristic approaches

Algorithm 1: Exhaustive Approach

    Data: Vertices in the graph V

    Result: $minVC$ An integer for the minumum vertex cover, VC an array of the verticies in the vertex cover

1 Let $PowerSet = $ list of all subsets of V;

2 $minVC=\infty$;

3 $VC=\{ \}$;

4 **for** $subset \in Powerset$ **do**

5 $subsetVC=subset.length$);

6   **if** $subset.isValid$ **and** $subsetVC < minVC$ **then**

7 $minVC=subsetVC$;

8 $VC= subset$;

9   **end**

10 **end**

Algorithm 2: Greedy-Heuristic Approach

    Data: V all vertices in graph G,E and all edges between the nodes of G

    Result: $minVC$ An integer for the minimum vertex cover, VC an array of the vertices in the vertex cover

1 $VC=\{ \}$;

2 $minVC=0$;

3 **while** E *is* **not** *empty* **do**

4 $vertex = $ MostUncoveredEdges(V, E) ;

5   **for** $edge \in E$ **do**

6   **if** $edge$ $connects$ to $vertex$ **then**

7 E.remove($edge$);

8 $minVC$+=1;

9 V.remove($vertex$);

10 VC.insert($vertex$);