1. Given a directed graph $G_i(V_i, E_i)$, find a minimum number of vertices, such that eliminating them from $G_i$, makes the graph cycle-free.

2. for $i \leq V_i$ (from 0 to $i$)
   for each vertex set $S$ with size $i$ ($|S| = i$)
   remove $S$ from $G_i$ and get $G_i'$
   DFS ($G_i'$)
   if $G_i'$ has no cycle
   minimum number $\leftarrow i$
   $FVS \leftarrow S$
   stop.

Complexity: $\sum_{i=0}^{IV_1} \binom{IV_1}{i} O(IV_1 + IE_1) = 2^{IV_1} O(IV_1 + IE_1)$ Exponential.

3.

i=0, 1 $G_i'$ has a cycle.
i=2 $S = \{1, 2, 3\}$ $G_i'$ has no cycle.
So the minimum number $= 2$
$FVS = \{1, 2\}$

I'll try to verify for all other subsets (by inspection) and we conclude that this is the optimum.
Heuristic Algorithm,

1. Select a vertex with the highest degree in \( G \).
2. Remove this from \( G \).
3. Apply DFS (\( G_1 \)).
4. (If \( G_1 \) has no cycle), stop else repeat 2 and 3.

Each iteration has at most \( |V| \) complexity:

- Selecting vertex: \( O(|V|) \)
- DFS: \( O(|V| + |E|) \)
- Total complexity: \( O(|V| \times (O(V) + O(V+E))) \) = \( O(|V| \times (|V| + |E|)) \).

Apply heuristic:

Select 1 or 2, then select 1.

\( FVS = \{1, 2, 3\} \) size 2

But had you selected vertex 1 (size 2):

It would have been the optimum.

Optimum \( FVS = \{1\} \)

Program & runtime results.