Problem type 1:

Answer the following question:

> *(See variants below)*

a. **BYH**

How many strongly connected components can a direct acyclic graph have?

b. **BYF**

How many topological sorts does a fully connected directed graph have?

c. **BYE**

What type of graph has the greatest number of topological sorts?

d. **BYD**

Given a directed graph, give a algorithm that finds the node that has the largest reach (find u such that $|rch(u)|$ is maximized).

e. **BYB**

Given a directed graph, give a algorithm that finds the node that has the smallest reach (find u such that $|rch(u)|$ is minimized).

f. **BYG**

You run DFS with pre/post numbering on a directed acyclic graph. You get the numbering for vertices u and v. You notice that the edge (u, v) can be classsified as a **forward** edge because of the relationship of the pre/post numberings.

Fill in the equality:

$$w < x < y < z$$

that must be true for a forward edge where

$$w, x, y, z, \in \{pre(u), post(u), pre(v), post(v)\}$$

g. **BYC**

You run DFS with pre/post numbering on a directed acyclic graph. You get the numbering for vertices u and v. You notice that the edge (u, v) can be classsified as a **backward** edge because of the relationship of the pre/post numberings.

Fill in the equality:

$$w < x < y < z$$

that must be true for a forward edge where

$$w, x, y, z, \in \{pre(u), post(u), pre(v), post(v)\}$$

h. **BYA**

You run DFS with pre/post numbering on a directed acyclic graph. You get the numbering for vertices u and v. You notice that the edge (u, v) can be classsified as a **cross** edge because of the relationship of the pre/post numberings.

Fill in the equality:

$$w < x < y < z$$

that must be true for a forward edge where

$$w, x, y, z, \in \{pre(u), post(u), pre(v), post(v)\}$$