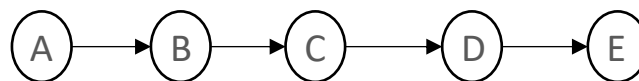


Reminder:

Lab test next week! Make sure you go to your allocated lab or you may not be allowed to sit the test.

Exercise 1: The following diagram represents a list of towns (A, B, C, D and E) and paths between them (the directed links)

You can get from **A** to **B**, but not from **B** to **A**. You can also get from **A** to **C**, by going **A** to **B**, and **B** to **C**.

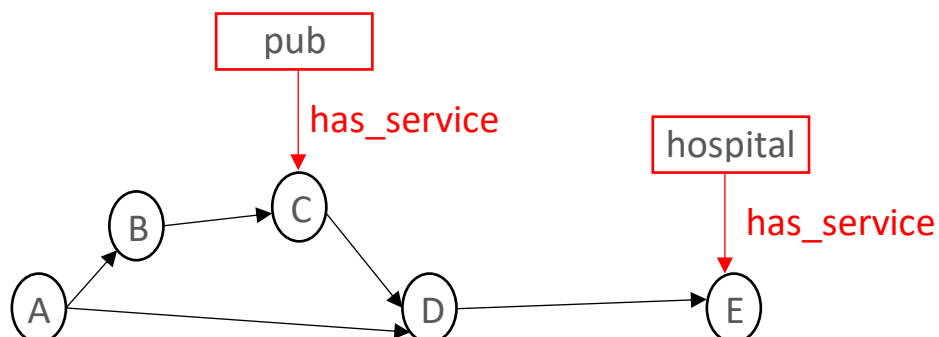


- Use a predicate **path(X,Y)** to represent that there is a path from X to Y to describe the problem above.
- Write a predicate **can_get(X,Y)** that returns true if you can get from X to Y following the paths. For example **can_get('A','C')** should return true, but **can_get('C','A')** should return false.

Exercise 2: Similarly to above, the following diagram represents a list of towns (A, B, C, D and E) and paths between them (the directed links). Some towns also have services, e.g. town C has a pub, town E has a hospital.

Similarly to above, use predicate **path(X,Y)** to represent that there is a path from X to Y. Also use predicate **has_service(X,Y)** to represent that town X has service Y.

Write a predicate **has_access(X, Y)** that return true if town X has access to service Y. For example, **has_access('A', pub)** should return true as one can travel A->B->C and C has a pub. However, **has_access('E', pub)** will return false as there is no path from E to a town that has a pub.



Exercise 3: Use the predicate **is_in(X, Y)** to represent that the Russian doll X is immediately nested in doll Y. For example **is_in('Bella', 'Anna')**



Write a predicate **contains(X,Y)** that should return true when doll X contains doll Y, either immediately or nested inside other dolls.

For example **contains ('Anna', 'Dora')** should return true, but **contains('Ella', 'Bella')** should return false.