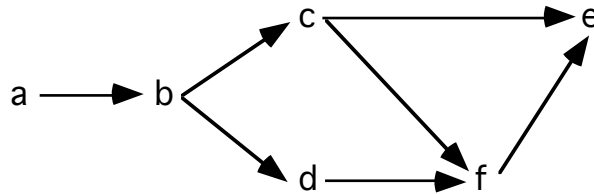


CM20019 – Prolog Lab – part 2

1. An acyclic directed graph is represented by the following facts:

```
arc(a, b).    arc(b, c).    arc(b, d).  
arc(c, f).    arc(d, f).    arc(c, e).  
arc(f, e).
```



a) Write two Prolog rules sufficient to define the relation `path(X, Z)` which holds when the graph contains a uniformly-directed path from `X` to `Z`. For instance, in the given graph there is a path from `b` to `f`.

Your program should refer only to the `path` and `arc` relations.

b) Try out your program for the queries (finding all solutions):

- i `?- path(b, f).`
- ii `?- path(b, Z).`
- iii `?- path(X, d).`

c) Enter and run queries asking if

- i there is a path from `a` to `f` that passes through `d`.
- ii there is a path of length 3 ending at `f` (assume each arc has unit length).
- iii there is a cycle in the graph.

2. The Peano numbers are `0`, `s(0)`, `s(s(0))`, ... etc. representing `0`, `1`, `2`, ... etc.

a) Write a Prolog program defining the relation `plus(X, Y, Z)` which holds when `X`, `Y` and `Z` are Peano numbers satisfying `X+Y=Z`. Use no relation other than `plus`.

b) Try out your program for the queries

- i `?- plus(s(0), s(s(0)), Z).`
- ii `?- plus(s(0), Y, s(s(s(s(0))))).`
- iii `?- plus(X, Y, s(s(s(s(0))))).`

c) Write a single rule defining the relation `odd(X)` in terms only of `plus`, where `odd(X)` expresses that `X` is an odd number. Test it with some queries.