# 5.3 BACKTRACKING



Date 05 04 2019

BACKTRACKING IS A STRATEGY OF GUESSING AT A SOLUTION AND RETRACING STEPS WHEN AN IMPASSE (DEADLOCK) IS REACHED.

## 5.3.1 SEARCHING FOR AN AIRLINE ROUTE

GOAL: FIND A PATH FROM SOME POINT TO DESTINATION POINT.

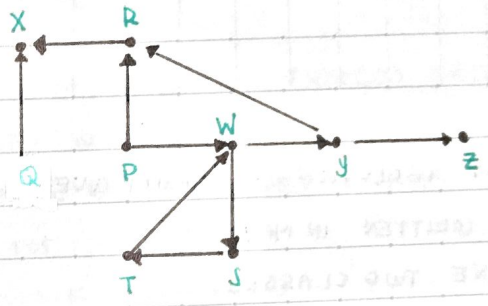
→ IN THIS CHAPTER WE WILL USE RECURSION TO SOLVE THIS.

HPAIR DATA WILL CONTAIN:

1. NAMES OF CITIES THAT HPAIR SERVES
2. PAIRS OF CITY NAMES (ORIGIN & DESTINATION)
3. PAIRS OF CITY NAMES (REQUEST TO FLY FROM SOME ORIGIN TO SOME DESTINATION)

HOW CAN WE SEARCH FLIGHT PATHS?

GIVEN: FROM CITY P TO CITY Z.




DIRECTED PATH:

P→W, W→y, y→Z

TO SOLVE THIS AN ALGORITHM MUST BE DEVELOPED THAT MIGHT INVOLVE A SINGLE OR MULTIPLE SEQUENCES OF FLIGHTS;

THIS IS AN EXHAUSTIVE SEARCH (SOL. WILL TRY EVERY POSSIBLE OF FLIGHTS UNTIL A SEQ. CAN BE FOUND OR NONE EXISTS.)

REFINED RECURSIVE SEARCH ALGORITHM

```
searchR (originCity: City, destinationCity : City): boolean {
    Mark originCity as visited
    if (originCity is destinationCity)
        Terminate — the destination is reached
    else
        for (each unvisited city C adjacent to originCity)
            searchR (C, destinationCity)
}
```

NOTE: WHEN SEARCHING TAKE INTO ACCOUNT THE POSSIBILITY THAT WRONG CHOICES WILL BE MADE

ALGORITHM MUST ELIMINATE POSSIBILITY OF CYCLING AS WELL AS BEING ABLE TO BACKTRACK WHEN A DEADEND OCCURS.

→ MAKE SURE THAT YOUR RECURSIVE SOLUTION EVENTUALLY REACHES A BASE CASE!