

## Heuristic Approach to TCG

- **Heuristic**
  - **Webster dictionary**
    - involving or serving as an aid to learning, discovery, or problem-solving by experimental and especially trial-and-error methods
- **Examples**
  - **Discussion**

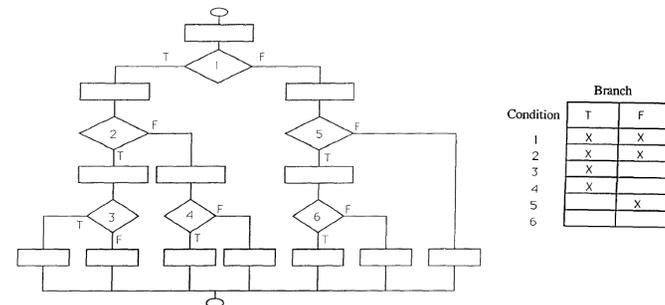
## Search Examples

- **Looking for a solution in a search space**
- **Known techniques**
  - **Depth-first**
  - **Breadth-first**
  - **Binary search for certain structures**
- **Others?**
  - **Large branching factor**
  - **Very deep**

## Do we need a Heuristic?

- **To generate test cases that achieve maximal branch coverage**

## Example Flow-chart



## Observations

- 1T, 1F, 2T, 2F have been covered
- 3T has been covered
- To cover 3F, can we tweak the test case for 3T?
- Also, what do we do when multiple test cases are available for tweaking?
  - Need a way to compare

## "Best Test Case"

- If (Exp) THEN \_\_ ELSE \_\_;
- Exp can be (LHS <op> RHS)
- The "goodness" of a test case t1  
$$\frac{|LHS(t1) - RHS(t1)|}{2 * \max(|LHS(t1)|, |RHS(t1)|)}$$
- Should we rely on only local information?
  - What are the risks?

## Better "Best Test Case"

$$G(t,D) = w * L(t,D) + (1-w) * P(t,D) \quad (2)$$

where:

G(t,D) : Goodness of test case t at condition D.

L(t,D) : Freedom space of t at D.

P(t,D) : Sum of freedom space reciprocals of t along the path toward D.

w : Weighting factor between L(t,D) and P(t,D),  $0 < w < 1$ .

L(t,D) is defined as in formula (1), and P(t,D) is defined as:

$$P(t,D) = \sum_{\text{all } D_i} 1 / (n * L(t,D_i)) \quad (3)$$

- Smallest value indicates the best test case