Celebrity Identification Problem

- **Solution:**
  - Optimal number of calls to "Does u knows v ?" in the worst case:

$$3(n-1)-\lfloor \log n \rfloor$$

  - Observation 2:
    - ➢ Some calls involving s in the verification phase may be redundant (already done during the elimination phase)
      - Maximum number of such could be $\lfloor \log n \rfloor$

✓Hint:

- ➢ Elimination phase may be implemented as a knock-out tournament
- ➢ A FIFO queue can be used to maximize the number (i.e., at least $\lfloor \log n \rfloor$) of matches played by the winner [A looser is never inserted back into the queue]
- ➢ Verification phase can be implemented such that it does not repeat the calls already made in the elimination phase

Celebrity Identification Problem

- **Solution:**
  - Optimal number of calls to "Does u knows v ?" in the worst case:

$$3(n-1) - \lfloor \log n \rfloor$$

Volume 14, number 3 INFORMATION PROCESSING LETTERS 16 May 1982

AN OPTIMAL ALGORITHM FOR SINK-FINDING

K.N. KING and Barbara SMITH-THOMAS
School of Information and Computer Science, Georgia Institute of Technology, Atlanta, GA 30332, U.S.A.

Received 16 November 1981; revised version received 18 February 1982

Graph algorithm, sink, Anderaa–Rosenberg conjecture, tournament, oracle

1. Introduction

The problem of sink-finding (determining whether a directed graph, represented by an adjacency matrix, has a sink – a vertex with no outgoing edges and an incoming edge from every other vertex) first appeared as a counterexample to an early version of the Anderaa–Rosenberg conjecture [4]. Rosenberg had originally conjectured that, for P a 'natural' property of graphs, the problem of determining whether P

We think of the vertices of G as players in a tournament. A match between players i and j corresponds to accessing a_{ij}: if $a_{ij} = 0$, then player i 'wins' (that is, vertex j cannot be a sink, while vertex i is still a possibility), while if $a_{ij} = 1$, then player j 'wins'. The analogy is not perfect, since the result when i plays j may differ from the result of j playing i. However, this approach leads to a simple derivation of upper and lower bounds for the problem of sink-finding.

Our sink-finding algorithm has two phases. In