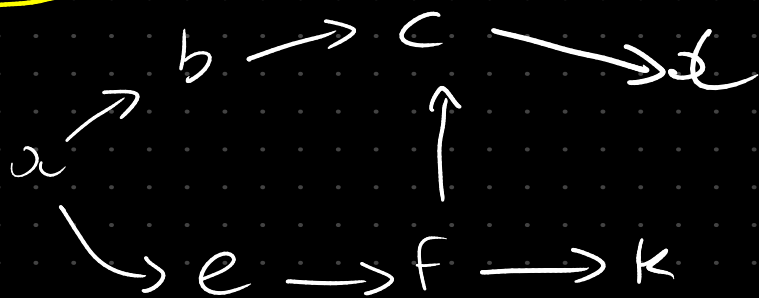


11/03/2024

LEZ 3 | FUNZIONAMENTO DI PROLOG

ESERCIZIO



DESCRIVERE GRAFO IN PROLOG E DESCRIVERE UN PATH (PERCORSO).

PROCEDIMENTO

- FATTI = TUTTI GLI EDGE NEL GRAFO
 $e(x, y) \dots$ ETC.

- $PATH(X, Y) \rightarrow$ IDEA

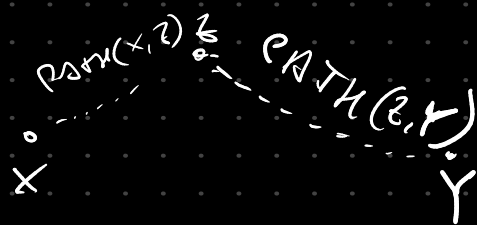
con CASO BASE

$PATH(X, Y) :-$
 $e(x, y)$

e CASO INDUTTIVO

$PATH(X, Y) :-$
 $PATH(X, Z),$
 $PATH(Z, Y).$

$PATH(X, Y)$




N.B. 2 REGOLE U = NONE
SONO DISTINTI

↓
QUANDO IL CODICE RIESEGUE
1 L'ALTRO NON VIENE
ESEGUITO, COME MAI?

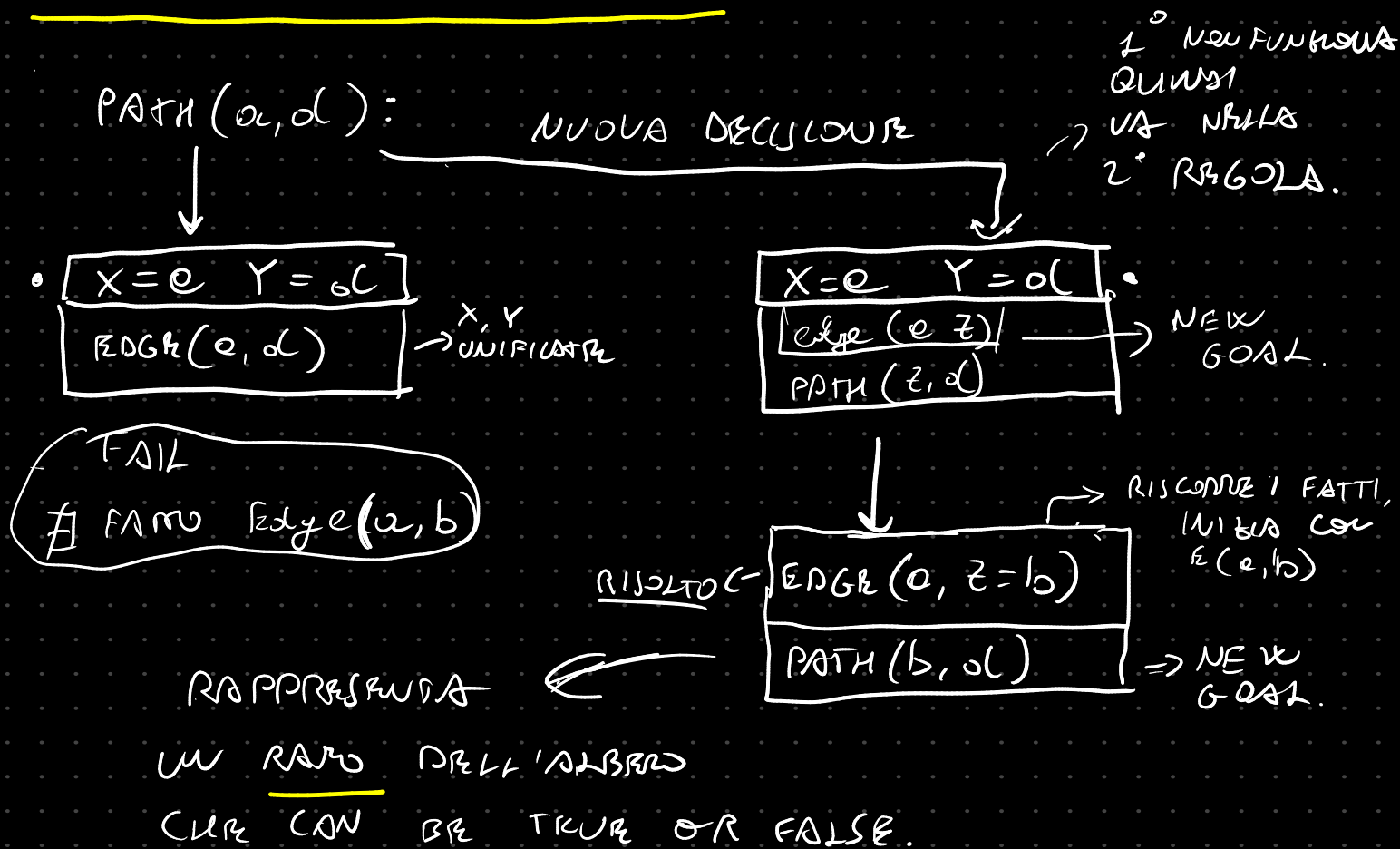
MA COSA FA IL CODICE?

ALGORITMO DI RISOLUZIONE

DURING THE QUERY, IL QUERY DIVENTA UN GOAL.
POI IL PROGRAMMA SCORRE DAL TOP FINO AL BASED
I FATTI, E SE  UNA REGOLA VA DA SINISTRA
VERSO DESTRA.

DURANTE LA RICERCA DEL GOAL, CREA
UN ALBERO DELLE DECISIONI. OVE AD OGNI NUOVA
RICHIESTA DI QUERY, (O NEW ASS. VARIABLE) RITORNA
INDIETRO E RICOMINCIA.

PRENDIAMO PATH COMPLESSIVE :



QUINDI:

- AD OGNI RICERCA OF GOAL
- SCORRE I FATTI E REGOLE (↓ X FATTI REGOLE, → X REG.)
 - APPLICA IL BACKTRACKING → TORNA INDIETRO E FA NUOVO TENTATIVO
 - CERCA UNA SEQUIENZA DI GOAL, SE NECESSARIO
- GOAL = RARO DELL' ALBERO DI DEC.
- NUOVO RARO ALBERO DECISIONE = NUOVO GOAL DA P

VEDENDO COME SI COMPORTA, NOTIAMO CHE

L'ORDINE È MOLTO IMPORTANTE.