

Лабораторна робота № 6

Тема: Представлення та обробка даних у Пролозі: графи та дерева

Мета роботи: набуття навичок роботи з різними структурами даних у Пролозі.

Порядок виконання роботи

Завдання 1

Створити предикати, що дозволять видалити довільну вершину на дереві, видалити піддерево починаючи з указаної вершини, якщо вона існує на дереві.

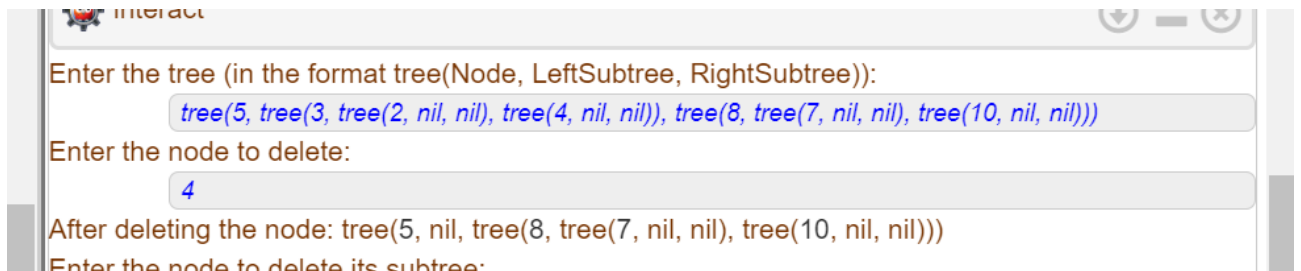


Рисунок 1 – Виконання завдання

Код програми:

`delete_node(Tree, Node, ResultTree) :-`

`delete_node_helper(Tree, Node, _, ResultTree).`

`delete_node_helper(tree(Node, Left, Right), Node, Left, Right).`

`delete_node_helper(tree(Root, Left, Right), Node, NewLeft, tree(Root, UpdatedLeft, Right)) :-`

`delete_node_helper(Left, Node, NewLeft, UpdatedLeft).`

`delete_node_helper(tree(Root, Left, Right), Node, tree(Root, Left, UpdatedRight), NewRight) :-`

`delete_node_helper(Right, Node, UpdatedRight, NewRight).`

`delete_subtree(Tree, Node, ResultTree) :-`

`delete_subtree_helper(Tree, Node, _, ResultTree).`

```
delete_subtree_helper(tree(Node, _, _), Node, nil, nil).
```

```
delete_subtree_helper(tree(Root, Left, Right), Node, NewLeft, tree(Root,  
UpdatedLeft, Right)) :-
```

```
    delete_subtree_helper(Left, Node, NewLeft, UpdatedLeft).
```

```
delete_subtree_helper(tree(Root, Left, Right), Node, tree(Root, Left, UpdatedRight),  
NewRight) :-
```

```
    delete_subtree_helper(Right, Node, UpdatedRight, NewRight).
```

```
read_tree(Tree) :-
```

```
    write('Enter the tree (in the format tree(Node, LeftSubtree, RightSubtree)): '),
```

```
    read(Tree).
```

```
print_tree(nil) :-
```

```
    write('nil').
```

```
print_tree(tree(Node, Left, Right)) :-
```

```
    write('tree('),
```

```
    write(Node),
```

```
    write(', '),
```

```
    print_tree(Left),
```

```
    write(', '),
```

```
    print_tree(Right),
```

```
    write(')').
```

```
interact :-
```

```
    read_tree(Tree),
```

```
    write('Enter the node to delete: '),
```

```
    read(NodeToDelete),
```

```
    delete_node(Tree, NodeToDelete, ResultTree),
```

```

write('After deleting the node: '),
print_tree(ResultTree),
nl,
write('Enter the node to delete its subtree: '),
read(NodeToDeleteSubtree),
delete_subtree(ResultTree, NodeToDeleteSubtree, FinalTree),
write('After deleting the subtree: '),
print_tree(FinalTree).
:- interact.

```

Завдання 2

Граф задано у вигляді БД дуг. Знайти найдовший шлях між заданими вершинами на графі з використанням пошуку в ширину.

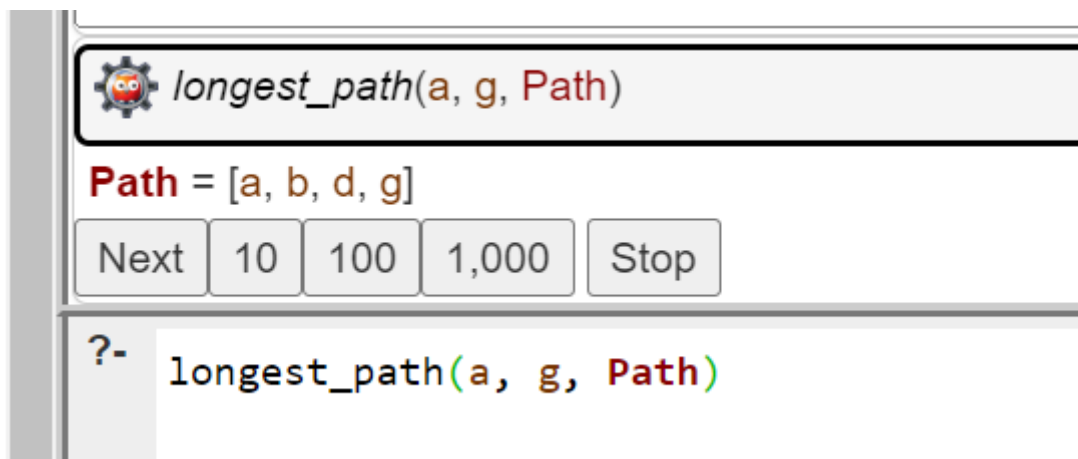


Рисунок 2 – Виконання завдання

Код програми:

```

arcs(a, b, 2).
arcs(a, c, 3).
arcs(b, d, 5).
arcs(b, e, 4).
arcs(c, f, 7).
arcs(d, g, 6).
arcs(e, g, 3).
arcs(f, g, 8).

```

```
longest_path(Start, Goal, Path) :-  
    dfs([[Start]], Goal, ReversePath),  
    reverse(ReversePath, Path).  
dfs([[Goal|Path]|_], Goal, [Goal|Path]).  
dfs([Path|Paths], Goal, FinalPath) :-  
    extend(Path, NewPaths),  
    append(Paths, NewPaths, Paths1),  
    dfs(Paths1, Goal, FinalPath).  
extend([Node|Path], NewPaths) :-  
    findall([NewNode, Node|Path],  
        (arcs(Node, NewNode, _),  
         \+ member(NewNode, [Node|Path])),  
        NewPaths).
```

Висновок: під час виконання лабораторної роботи було набуто навички роботи з різними структурами даних у Пролозі.