Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп'ютерних технологій

Кафедра радіоелектронних та комп'ютерних систем

ЗВІТ

про виконання лабораторної роботи №4

"Пошук з поверненням"

Виконав студент

групи Феі- 54

Вегера Северин

Викладач:

доц. Бабич О. Й.

Львів — 2018

Короткі теоретичні відомості:

Пошук з поверненням (**backtracking**) - це один з основних прийомів пошуку розвязань поставленого завдання в ПРОЛОЗІ. Виконуючи пошук, ПРОЛОГ може зіткнутися з необхідністю вибору між альтернативними шляхами. Тоді він ставить маркер у місця розвилки (точка відкату) і вибирає першу підціль. Якщо вона не виконується, то ПРОЛОГ повертається в точку відкату і переходить до наступної підцілі.

Середовище Visual Prolog дозволяє використовувати відлагоджувач для покрокового виконання програми. Відлагоджувач працює з відкомпільованим кодом. У вихідному коді можна ставити точки зупинки і виконувати програму по кроках. У режимі покрокового виконання програми можна переглядати значення змінних і вміст затверджених фактів.

На cьогоднішній день Пролог є живою мовою, яка активно розвивається, хоча і не має такого широкого розповсюдження як C++, С# і створені на їх основі інтегровані середовища розробки. З початку своєї появи в 70-х рр. ХХ-го століття система програмування на Пролозі існувала як **інтерпретатор**. У 1988 році з'явилася система Turbo Prolog, а потім і Visual Prolog, що включають **компілятор** і **інтегроване середовище** розробки програм. Розроблена не так давно 7-ма версія Visual Prolog є **об'єктно-орієнтованим середовищем** розробки логічних програм. Математична основа і механізм роботи Прологу однакові для інтерпретатора і компілятора; разом з тим у побудові програм є деякі суттєві відмінності. Класичні праці, що пояснюють механізм роботи і області застосування Прологу, написані для інтерпретатора. Тому, незважаючи на те, що компілятори витіснили інтерпретуючі системи ми будемо розбирати деякі питання програмування як для компіляторів, так і для інтерпретаторів. При цьому система програмування з **інтерпретатором** називається **традиційною системою Пролог**, а **з компілятором - Visual Prolog (VIP)**. Якщо мова йде про загальні питання, то називається просто Пролог.

**Інтерпретатор** мови програмування (interpreter) — програма чи технічні засоби, необхідні для виконання інших програм, вид транслятора, який здійснює пооператорну (покомандну, полінійчату) обробку, перетворення у машинні коди та виконання програми або запиту (на відміну від компілятора, який транслює у машинні коди всю програму без її виконання).

Постановка задачі:

База даних містить наступні факти:

захоплюється ("Коля", гітара).

захоплюється ("Оля", скрипка).

захоплюється ("Діма", плавання).

захоплюється ("Таня", теніс).

спорт (плавання).

спорт (теніс).

муз\_інстр (скрипка).

муз\_інстр (гітара).

Завдання:

1.а) скласти правило спортсмен і визначити, хто захоплюється спортом;

1.б) простежити за пошуком розвязання за допомогою відлагоджувача;

1.в) побудувати цільове дерево пошуку з поверненням.

2.а) скласти правило музикант і визначити, хто захоплюється музикою;

2.б) простежити за пошуком розвязання за допомогою відлагоджувача;

2.в) побудувати цільове дерево пошуку з поверненням.

Остаточний код задачі:

implement main

open core

domains

імя = string.

захоплення = string.

class predicates

захоплюється : (імя, захоплення) nondeterm anyflow.

спорт : (захоплення) nondeterm anyflow.

муз\_інстр : (захоплення) nondeterm anyflow.

спортсмен : (імя) nondeterm anyflow.

музикант : (імя) nondeterm anyflow.

clauses

захоплюється ("Коля", "гітара").

захоплюється ("Оля", "скрипка").

захоплюється ("Діма", "плавання").

захоплюється ("Таня", "теніс").

спорт ("плавання").

спорт ("теніс").

муз\_інстр ("скрипка").

муз\_інстр ("гітара").

спортсмен (X):-захоплюється (X, Y), спорт(Y).

музикант (X):-захоплюється (X, Y), муз\_інстр(Y).

clauses

run() :-

console::init(),

stdIO::writef("Займаються спортом:\n"),

спортсмен(X), stdIO::writef(X),stdio::nl,

fail.

run() :-

console::init(),

stdIO::writef("\nЗаймаються музикою:\n"),

музикант(Y), stdIO::writef(Y),stdio::nl,

fail.

run() :-

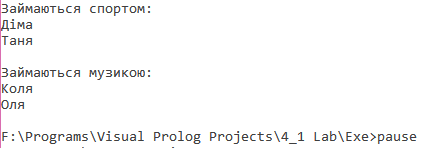
stdIO::writef("").

end implement main

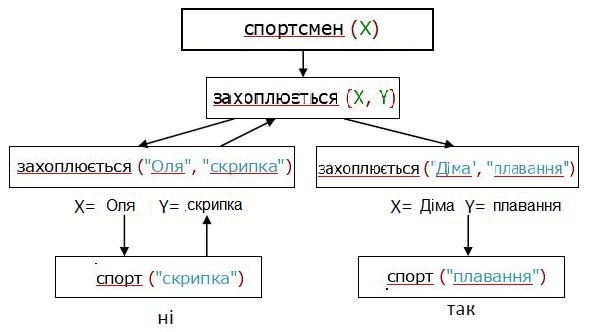
goal

console::runUtf8(main::run).

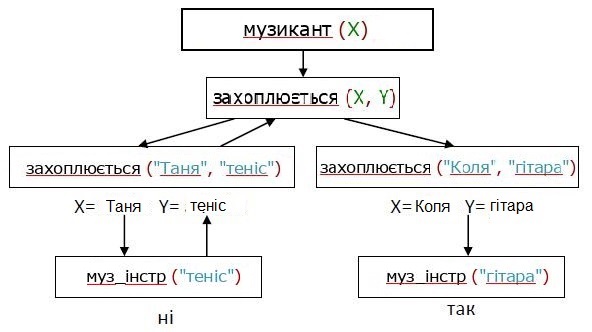
Результати:



Цільове дерево пошуку розвязання задачі "Спортсмен":



Цільове дерево пошуку розвязання задачі "Музикант":



Висновок:

На цій лабораторній роботі, я ознайомився із програмними можливостями середовища Visual Prolog і створив проект у якому використав один з основних прийомів пошуку розвязань поставленого завдання в ПРОЛОЗІ -- пошук з поверненням (backtracking). Склав правила спортсмен і музикант. Визначив хто займається спортом а хто займається музикою. Реалізував код, який виконує необхідні запити.