Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій

Кафедра програмних систем і технологій

ЗВІТ

з лабораторної роботи № 1

Тема: “Автоматизована система платежів”

Дисципліна «Об’єктно-орієнтоване конструювання програм»

Підготував:

студент гр. ІПЗ-23(1)

Гоша Давід Олександрович

Дата : 30.01.22

Перевірила:

доц. Зубик Л.В.

# Київ – 2022

**Тема:** Об’єктно-орієнтоване програмування у різних мовах програмування (COOL&CLIPS).

**Мета роботи**: ознайомитися з об’єктноорієнтованим підходом у різних мовах програмування, зокрема COOL&CLIPS.

**Умова**

1. Розробити програмний код для вирішення поставленої задачі. Середовище для розроблення – CLIPS. Мова програмування – COOL. Дозволяється використання онлайн-сервісів у процесі розроблення.

**Варіант 4**

На допит були викликано четверо підозрюваних в пограбуванні: А, B, С і D. Неспростовними доказами доведено, що принаймні один з них винен і що ніхто, окрім А, B, С і D, в пограбуванні не брав участь. Крім того, вдалося встановити наступне: 1) А безумовно не винен. 2) Якщо B винен, то у нього був рівно один співучасник. 3) Якщо С винен, то у нього було рівне два співучасники. Інспектору Скотланд-Ярду було особливо важливо взнати, винен або не винен D, оскільки D був 92 небезпечним злочинцем. На щастя, приведених вище фактів достатньо, щоб встановити винність або невинність підозрюваного D.

**Хід роботи:**  
Структуруємо отриману інформації з умови. У нас є 4 персони (A , B , C , D) Де A – Безумовно невинен (з умови). B – має одного напарника. С – має двох. А D – невідомо.

Для початку створимо факт та зазначимо ім’я та кількість співучасників з підозрюваних. Далі додамо правило . що А – безумовно невинен.

У другому правилі зазначимо два факти , але щоб впевнитися у правоті твердження , що D – винен , ми маємо підтвердити третій факт , що згодом дасть позитивний результат.

Хід логіки доволі простий. Якщо А – безумовно невинен , мі віключаємо його зі списку підозрюємих , тому при наступному правилі , зробило два припущення , що винен С або D. Так як ми не впевнині що винен D , то нам треба перевірити твердження про С.

Так як за умовою С має 2 напарники і це не А, то залишаються тільки D та B.

Тому B Винен у будь якому випадку. Ми перевіряємо , чи існують 2 факти, а саме припущення що B – винен , та що С винен і його співучасник В.

**Код**

(deffacts Suspects

(Sus A 0)

(Sus B 1)

(Sus C 2)

)

(defrule Suspects1

(Sus ?name ?partners)

(test (= ?partners 0))

=>

(assert (sus isnt ?name )

))

(defrule Suspects2

(Sus ?name ?partners)

(test (= ?partners 1))

=>

(bind ?n sus is C)

(bind ?k sus is D )

(assert (sus is C )(sus is D )

))

(defrule Suspends\_check

(Sus ?name ?partners )

(forall(Sus ?name ?partners)

(sus is C ))

=>

(assert (sus is B or C , D )

(printout t "B or C , D to jail " crlf ))

)

(defrule Suspects5

(forall(Sus ?name ?partners)

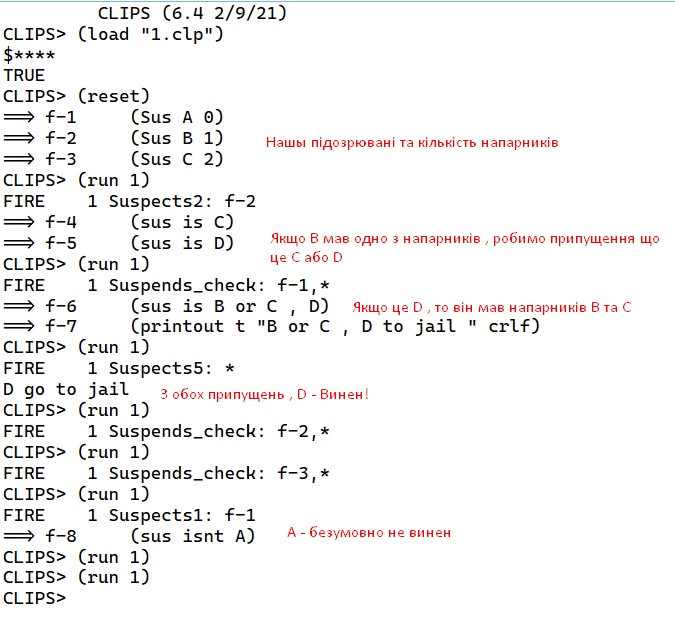
(sus is D )

(sus is B or C , D ) )

=>

(printout t "D go to jail" crlf ))

**Скриншоти Виконання**



Висновки:

Сьогодні ми навчилися використовувати мову програмування CLIPS , для розробки експертних систем та розробили застосунок, що вирішує логічну задачу з підозрюваними. Це гнучка мова, яку можна інтегрувати у проєкти на ішних мовах , зокрема C#. CLIPS зручний тим, що принцип написання застосунків на ньому схожий до людського мишлення , напроти традиційних мов , де дії упорядкованя алгоритмічно.

Зі складностей я відмічу незвичайний синтаксис , більш схожий на Lisp. Та зміна звичної логіки підходу до написання Пз.