МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

Кафедра «Информационные системы»

РАСЧЕТНО-ГРАФИЧЕСКАЯ РАБОТА

по дисциплине «Методы и системы искусственного интеллекта»

вариант №17

Выполнил: ст.гр. ИС/б-20-1-о

Лысенко И.Д.\_\_\_\_-\_\_\_\_\_\_

Проверил: Забаштанский А.К.\_\_\_

Севастополь

2024

1. **Вариант задания**

**Вариант № 17**

Разработать на языке Пролог реляционную базу данных (не меньше 5 фактов), соответствующую приведенной таблице, составить 5 правил, удовлетворяющих условиям: 1) проверить, верно ли, что в базе данных есть хотя бы один продукт, стоимость которого больше 300 рублей, но меньше 1000 рублей; 2) проверить, верно ли, что в базе данных есть хотя бы один продукт, стоимость которого больше 1000 рублей или меньше 300 рублей; 3) проверить, верно ли, что в базе данных есть хотя бы один продукт, у которого порядковый номер и номер вольера совпадают (если есть, то вывести его название); 4) вывести название всех продуктов, которые имеют вес, заданным пользователем; 5) проверить, есть ли в базе данных два продукта одного веса и одной стоимости.

К каждому правилу реализовать характерные запросы (не менее двух).

Таблица 1 – Потребление продуктов животными зоопарка за одни сутки

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Название продукта | Вес продукта | Стоимость продукта | Вид животного | Кличка животного | № вольера |

1. **Ход работы**

Были составлены факты и правила, удовлетворяющие заданию варианта (листинг 1).

Листинг 1 – Полный код программы на языке Пролог

:-dynamic(consumedProduct/7).

% consumedProduct(N, Name, Weight, Cost, AnimalSpec, Nickname, CageNumber)

consumedProduct(1, meat, 3, 250, tiger, bobik, 6).

consumedProduct(2, pickles, 1, 60, zebra, oleg, 2).

consumedProduct(3, watermelon, 3, 300, capybara, ignat, 9).

consumedProduct(4, fish, 5, 430, carakal, shlyopa, 6).

consumedProduct(5, grass, 3, 300, cow, muza, 1).

% 1) check if at least one with cost > 300 and cost < 1000

less\_and\_more\_cost :-

consumedProduct(\_, \_, \_, Cost, \_, \_, \_),

Cost > 300,

Cost < 1000.

% 2) check if at least one with cost < 300 or cost > 1000

less\_or\_more\_cost :-

consumedProduct(\_, \_, \_, Cost, \_, \_, \_),

(Cost < 300; Cost > 1000).

% 3) check if at least one where number equals cage number

number\_equals\_cageNumber :-

consumedProduct(Number, Name, \_, \_, \_, \_, CageNumber),

Number = CageNumber,

write('Such record exists: Name is '), write(Name), nl,

!;

write('Not Found').

number\_equals\_cageNumber.

% 4) get products list by input weight

get\_products\_by\_weight :-

write('Enter weight: '), nl, read(Weight),

findall(consumedProduct(N, Name, Weight, Cost, AnimalSpec, Nickname, CageNumber), consumedProduct(N, Name, Weight, Cost, AnimalSpec, Nickname, CageNumber), Products),

(Products = [] ->

write('No records found.');

write('Here is the list of products with weight equal to: '), write(Weight), write(':'), nl, nl,

print\_products(Products)

).

print\_products([]).

print\_products([consumedProduct(N, Name, Weight, Cost, AnimalSpec, Nickname, CageNumber)|Rest]) :-

write('Number: '),

write(N), nl,

write('Name of the product: '),

write(Name), nl,

write('Weight: '),

write(Weight), nl,

write('Cost: '),

write(Cost), nl,

write('Animal spec: '),

write(AnimalSpec), nl,

write('Animal nickname: '),

write(Nickname), nl,

write('Animal cage number: '),

write(CageNumber), nl,

get0(C),

print\_products(Rest).

% 5) check if there are two records with equal weight and cost

equal\_weight\_and\_cost :-

consumedProduct(N1, \_, Weight1, Cost1, \_, \_, \_),

consumedProduct(N2, \_, Weight2, Cost2, \_, \_, \_),

N1 \= N2,

Weight1 = Weight2,

Cost1 = Cost2,

write('Such records found, their numbers are '), write(N1), write(' and '), write(N2), nl,

!;

write('Such Records Not Found').

Разработанный код был протестирован.

Была протестирована функция проверки наличия хотя бы одного продукта с ценой более 300 и менее 1000. Затем были изменены данные записей для проверки работы функции в иных условиях. Результат представлен на рисунках 1-2.

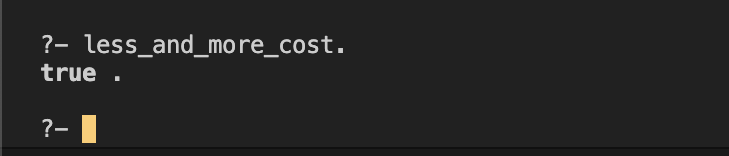


Рисунок 1 - Тестирование функции 1

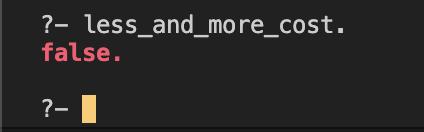


Рисунок 2 - Тестирование функции 1

Далее была протестирована функция проверки наличия хотя бы одного продукта с ценой менее 300 или более 1000. Затем были изменены данные записей для проверки работы функции в иных условиях. Результат представлен на рисунках 3-4.

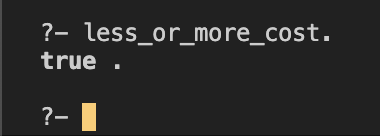


Рисунок 3 - Тестирование функции 2

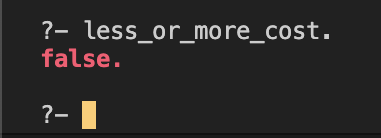


Рисунок 4 - Тестирование функции 2

Далее была протестирована функция проверки наличия продукта с Одинаковыми номерами и номерами вольеров. Затем были изменены данные записей для проверки работы функции в иных условиях. Результат представлен на рисунках 5-6.

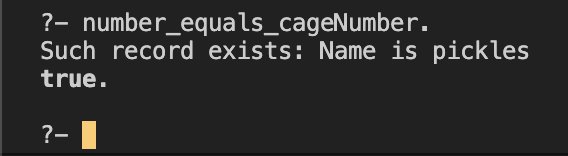


Рисунок 5 - Тестирование функции 3

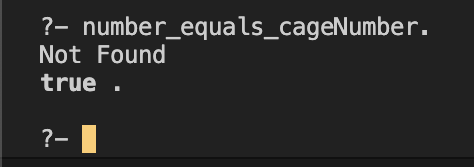


Рисунок 6 - Тестирование функции 3

Далее была протестирована функция получения списка продуктов по введённому пользователем весу. Затем были изменены данные записей для проверки работы функции в иных условиях. Результат представлен на рисунках 7-8.

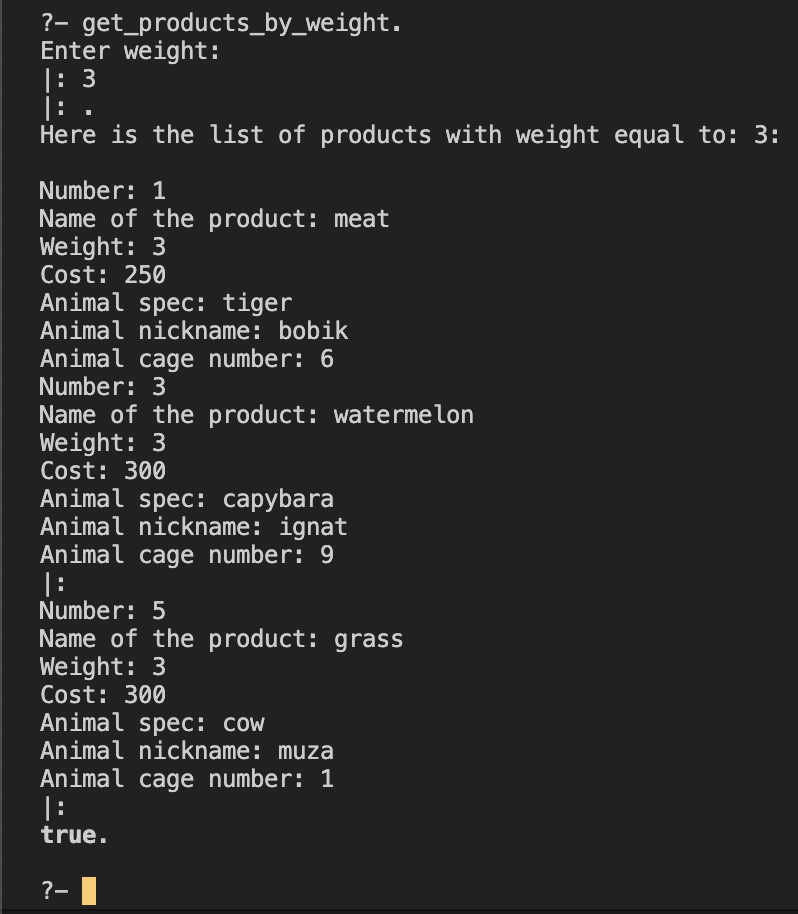


Рисунок 7 - Тестирование функции 4

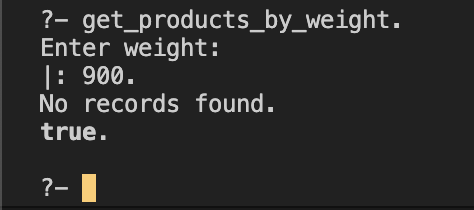


Рисунок 8 - Тестирование функции 4

Далее была протестирована функция проверки наличия двух продуктов с одинаковыми весами и ценами. Затем были изменены данные записей для проверки работы функции в иных условиях. Результат представлен на рисунках 9-10.

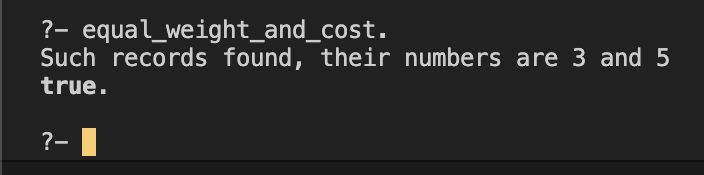


Рисунок 8 - Тестирование функции 5

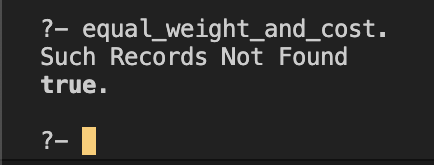


Рисунок 9 - Тестирование функции 5

Таким образом, можно сделать вывод о том, что разработанная программа работает корректно.

1. **Выводы**

В ходе выполнения расчётно-графической работы были закреплены базовые навыки составления программ на языке Пролог и умение ориентироваться в синтаксисе языка. Была составлена программа, выполняющая поставленные по варианту задания задачи.