|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | «Головной учебно-исследовательский и методический центр профессиональной реабилитации лиц с ограниченными возможностями здоровья (инвалидов)» |
| КАФЕДРА | «Системы обработки информации и управления» (ИУ5) |

***ДОМАШНЕЕ ЗАДАНИЕ***

***ПО КУРСУ ПИКЯП***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ5Ц-51Б |  |  |  | Р.Р. Чупанова |
|  | (группа) |  | (подпись, дата) |  | (И.О. Фамилия) |
| Преподаватель |  |  |  |  | А.Н. Нардид |
|  |  |  | (подпись, дата) |  | (И.О. Фамилия) |
|  |  |  |  |  |  |

*2024 г.*

# Задача

1. Выберите язык программирования (который Вы ранее не изучали) и (1) напишите по нему реферат с примерами кода или (2) реализуйте на нем небольшой проект (с детальным текстовым описанием).
2. Реферат (проект) может быть посвящен отдельному аспекту (аспектам) языка или содержать решение какой-либо задачи на этом языке.
3. Необходимо установить на свой компьютер компилятор (интерпретатор, транспилятор) этого языка и произвольную среду разработки.
4. В случае написания реферата необходимо разработать и откомпилировать примеры кода (или модифицировать стандартные примеры).
5. В случае создания проекта необходимо детально комментировать код.
6. При написании реферата (создании проекта) необходимо изучить и корректно использовать особенности парадигмы языка и основных конструкций данного языка.
7. Приветствуется написание черновика статьи по результатам выполнения ДЗ. Черновик статьи может быть подготовлен группой студентов, которые исследовали один и тот же аспект в нескольких языках или решили одинаковую задачу на нескольких языках.

# Код решения

Программа 1:

quadratic\_roots(A, B, C, Roots) :-

( A =:= 0

-> ( B =\= 0

->

Root is -C / B,

Roots = [Root]

;

(C =\= 0 -> Roots = [] ; Roots = [inf])

)

;

Discriminant is B \* B - 4 \* A \* C,

( Discriminant > 0

-> Root1 is (-B + sqrt(Discriminant)) / (2 \* A),

Root2 is (-B - sqrt(Discriminant)) / (2 \* A),

Roots = [Root1, Root2]

; Discriminant =:= 0

-> Root is -B / (2 \* A),

Roots = [Root]

; % Если дискриминант меньше 0

Roots = []

)

).

% ?- quadratic\_roots(1, -3, 2, Roots).

% Roots = [2.0, 1.0].

% ?- quadratic\_roots(1, 2, 1, Roots).

% Roots = [-1.0].

% ?- quadratic\_roots(1, 0, 1, Roots).

% Roots = [].

Программа 2:

edge(a, b).

edge(a, c).

edge(b, d).

edge(c, d).

edge(d, e).

edge(c, f).

path(Start, End, Path) :- % Путь между двумя вершинами

travel(Start, End, [Start], Path).

travel(Start, End, Visited, [Start|Path]) :-

edge(Start, Next),

\+ member(Next, Visited),

( Next = End

-> Path = [End]

; travel(Next, End, [Next|Visited], Path)

).

% ?- path(a, e, Path).

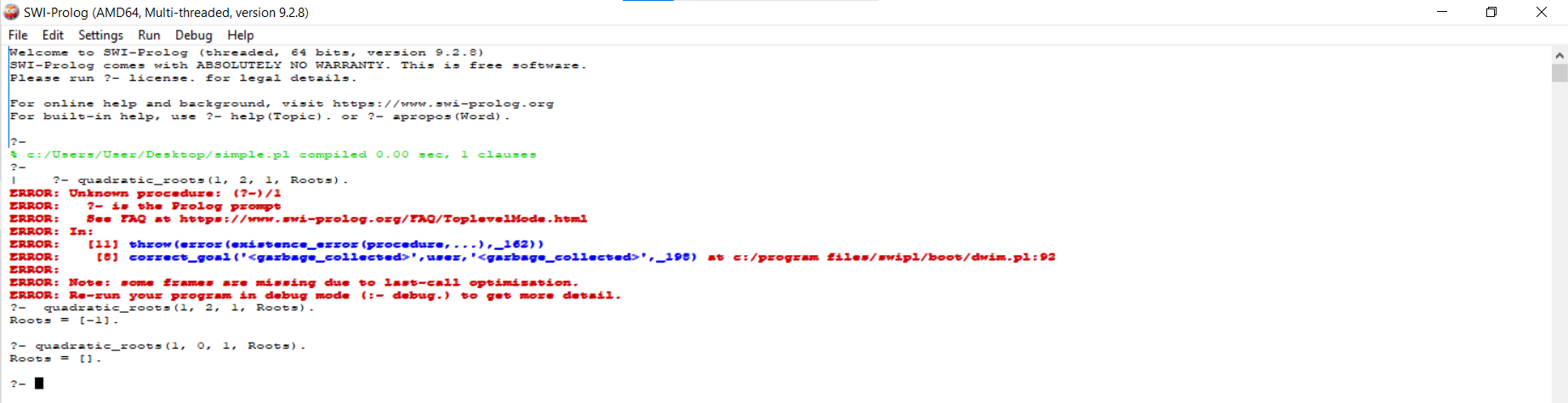
% Path = [a, c, d, e].

% ?- path(a, f, Path).

% Path = [a, c, f].

# Результат

Программа #1:



Программа #2:

