



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 «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 10

По дисциплине «Функциональное и логическое программирование»

Студент: Тимонин А. С.

Группа ИУ7-626

Преподаватель Толпинская Н. Б.

Москва.
2020 г.

Практическая часть

Ответить на вопросы (коротко):

1. Как организуется хвостовая рекурсия в Prolog?
2. Какое первое состояние резольвенты?
3. Каким способом можно разделить список на части, какие, требования к частям?
4. Как выделить за один шаг первые два подряд идущих элемента списка?
Как выделить 1-й и 3-й элемент за один шаг?
5. Как формируется новое состояние резольвенты?
6. Когда останавливается работа системы? Как это определяется на формальном уровне?

Используя хвостовую рекурсию, разработать, комментируя аргументы, эффективную программу, позволяющую:

1. Сформировать список из элементов числового списка, больших заданного значения;
2. Сформировать список из элементов, стоящих на нечетных позициях исходного списка (нумерация от 0);
3. Удалить заданный элемент из списка (один или все вхождения);
4. Преобразовать список в множество (можно использовать ранее разработанные процедуры).

Убедиться в правильности результатов

Для одного из вариантов ВОПРОСА и 1-ого задания составить таблицу, отражающую конкретный порядок работы системы:

Т.к. резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты! Для каждого запуска алгоритма унификации, требуется указать № выбранного правила и соответствующий вывод: успех или нет –и почему.

[illegible]

```

write("List[1, 2, 3, 4, 5, 6, 1, 2, 3, 7, 9, 10, 100] with nums above 3: \n"),
newListAboveNum([1, 2, 3, 4, 5, 6, 1, 2, 3, 7, 9, 10, 100], 3, List);
write("\nList[1, 3, 2, 1, 1, 1, 3, 3, 3, 2, 2, 2] with nums above 2: \n"),
newListAboveNum([1, 3, 2, 1, 1, 1, 3, 3, 3, 2, 2, 2], 2, List).

write("List[[1, 2, 3, 4, 5, 6, 7, 8] with oddp Index: \n"),
newListEvenPos(0, [1, 2, 3, 4, 5, 6, 7, 8], List);
write("\nList[0, 1, 0, 1] with oddp Index: \n"),
newListEvenPos(0, [0, 1, 0, 1], List).

write("Delete 3 in List[3, 3, 3] \n"),
deleteNumInList([3, 3, 3], 3, List);
write("\nDelete 0 in List[1, 0, 2, 0, 0, 3, 4] \n"),
deleteNumInList([1, 0, 2, 0, 0, 3, 4], 0, List).

write("Make set of List[5, 5, 5] \n"),
makeSet([5, 5, 5], Result);
write("\nMake set of List[3, 1, 1, 2, 1, 2] \n"),
makeSet([3, 1, 1, 2, 1, 2], Result).

```

```

[Inactive C:\Users\ANTONT~1\AppData\Local\Temp\goal$000.exe]
List[1, 2, 3, 4, 5, 6, 1, 2, 3, 7, 9, 10, 100] with nums above 3:
List=[4,5,6,7,9,10,100]

List[1, 3, 2, 1, 1, 1, 3, 3, 3, 2, 2, 2] with nums above 2:
List=[3,3,3,3]
2 Solutions

```

Рисунок 1. Тестирование newListAboveNum([mlist](#), [integer](#), [mlist](#))

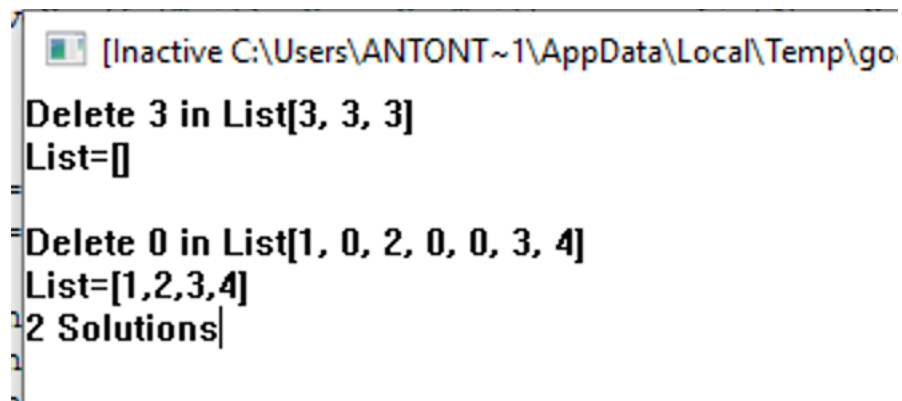
```

[Inactive C:\Users\ANTONT~1\AppData\Local\Temp\goal$000.exe]
List[[1, 2, 3, 4, 5, 6, 7, 8] with oddp Index:
List=[2,4,6,8]

List[0, 1, 0, 1] with oddp Index:
List=[1,1]
2 Solutions

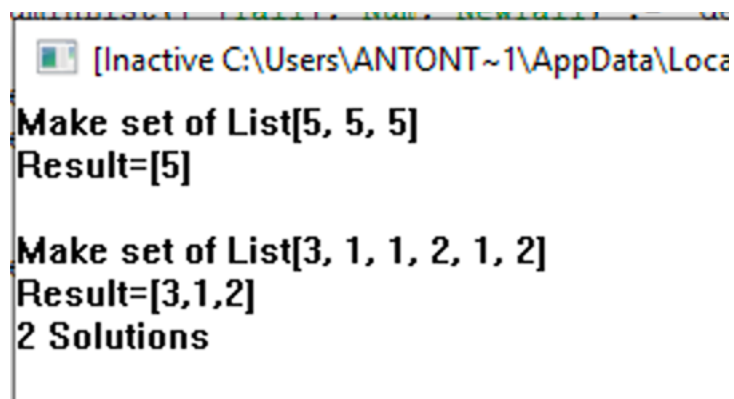
```

Рисунок 2. Тестирование newListEvenPos([integer](#), [mlist](#), [mlist](#))



```
[Inactive C:\Users\ANTONT~1\AppData\Local\Temp\go  
Delete 3 in List[3, 3, 3]  
List=[]  
Delete 0 in List[1, 0, 2, 0, 0, 3, 4]  
List=[1,2,3,4]  
2 Solutions
```

Рисунок 3. Тестирование deleteNumInList([mlist](#), [integer](#), [mlist](#))



```
[Inactive C:\Users\ANTONT~1\AppData\Local  
Make set of List[5, 5, 5]  
Result=[5]  
Make set of List[3, 1, 1, 2, 1, 2]  
Result=[3,1,2]  
2 Solutions
```

Рисунок 4. Тестирование makeSet([mlist](#), [mlist](#))

Формирование ответа

Таблица 1. deleteNumInList([1, 0, 2, 0, 3, 4], 0, List)

| № шага | Состояние резольвенты, и вывод: дальнейшие действия (почему?) | Для каких термов запускается алгоритм унификации: T1=T2 и каков результат | Дальнейшие действия: прямой ход или откат |
|--------|---------------------------------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------|
| 1 | deleteNumInList([1, 0, 2, 0, 3, 4], 0, List) | Подстановка Head = 1 , Tail = [0, 2, 0, 3, 4], Num = 0, = List = [Head NewTail] | Прямой ход |
| 2 | Head <> Num deleteNumInList(Tail, Num, NewTail) | 1 <> 0 Успех | Прямой ход |
| 3 | deleteNumInList(Tail, Num, NewTail) | Подстановка Head = 0, Tail = [2, 0, 3, 4], Num = 0, NewTail = [1 NewTail] | Прямой ход |
| 4 | Head <> Num deleteNumInList(Tail, Num, NewTail) | 0 <> 0 Неудача | Откат |
| 5 | deleteNumInList(Tail, Num, NewTail) | Подстановка Head = 2 , Tail = [0, 3, 4], Num = 0, NewTail = NewTail | Прямой ход |
| 6 | Head <> Num deleteNumInList(Tail, Num, NewTail) | 2 <> 0 Успех | Прямой ход |
| 7 | deleteNumInList(Tail, Num, NewTail) | Подстановка Head = 0, Tail = [3, 4], Num = 0, NewTail = [2 NewTail] | Прямой ход |
| 8 | Head <> Num deleteNumInList(Tail, Num, NewTail) | 0 <> 0 Неудача | Откат |
| 9 | deleteNumInList(Tail, Num, NewTail) | Подстановка Head = 3 , Tail = [4], Num = 0, NewTail = NewTail | Прямой ход |
| 10 | Head <> Num deleteNumInList(Tail, Num, NewTail) | 3 <> 0 Успех | Прямой ход |
| 11 | deleteNumInList(Tail, Num, NewTail) | Подстановка Head = 4 , Tail = [], Num = 0, NewTail = [3 NewTail] | Прямой ход |
| 12 | Head <> Num deleteNumInList(Tail, Num, NewTail) | 4 <> 0 Успех | Прямой ход |
| 13 | deleteNumInList(Tail, Num, NewTail) | Подстановка Tail = [], Num = 0, NewTail = [4 NewTail] | Прямой ход |
| 14 | deleteNumInList([], _, Result) | Подстановка Result = [1, 2, 3, 4] | Прямой ход |
| 15 | Пусто | Результат List = [1, 2, 3, 4] | Откат |

Теоретическая часть

Как организуется хвостовая рекурсия в Prolog?

Хвостовая рекурсия в Пролог организуется при помощи правила, которое обращается к тому же правилу.

Какое первое состояние резольвенты?

Простой вопрос

Каким способом можно разделить список на части, какие, требования к частям?

Prolog существует более общий способ доступа к элементам списка. Для этого используется метод разбиения списка на начало и остаток. Начало списка – это группа первых элементов, не менее одного. Остаток списка – обязательно список (может быть пустой). Для разделения списка на начало, и остаток используется вертикальная черта (|) за последним элементом начала.

Как выделить за один шаг первые два подряд идущих элемента списка?

```
lis([A | [B | C]]) :- write(A),  
                     write(B),  
                     write(C),  
                     !.
```

```
lis([1, 2, 3, 4, 5]).
```

```
A = 1  
B = 2  
C = [3, 4, 5]
```

Как выделить 1-й и 3-й элемент за один шаг?

```
lis([A | [_ | [B | C]]) :- write(A),  
                          write(B),  
                          write(C),  
                          !.
```

```
lis([1, 1000, 3, 4, 5, 6, 7, 8, 9]).
```

A = 1
B = 3
C = [4, 5, 6, 7, 8, 9]

Как формируется новое состояние резольвенты?

При возврате отменяется последняя уже выполненная редукция (восстанавливается предыдущее состояние резольвенты) и система выполняет ре- конкретизацию переменных, которые были конкретизированы на предыдущем шаге.

Когда останавливается работа системы? Как это определяется на формальном уровне?

Работа интерпретатора завершается либо когда список инструкций опустеет, либо когда произойдет какая-либо ошибка во время выполнения инструкции.

Исправления 1

Тимонин А. С. ИУ7-626 В письме правильно указывайте номер, иначе не буду открывать!

— Извините пожалуйста!

В каком фрагменте программы сформулировано знание?

— Знания о предметной области выражаются на языке Prolog в виде предложений, называемых утверждениями- CLAUSES. Каждое утверждение заканчивается точкой и описывает какое-либо отношение, свойство, объект или закономерность. Структура утверждения имеет одну из форм:

- <заголовок>. - факт
- <заголовок> :- тело. - правило, где заголовок является предикатом и полностью характеризует описываемое отношение.

Каковы назначение и результат использования алгоритма унификации? Четкого ответа нет!

— Назначение алгоритма унификации заключается в том, чтобы связать переменные из вопроса со значениями, находящимся в правилах и фактах.

Goal

$p1 = \text{person}(\langle \text{Anton} \rangle, \langle \text{Timonin} \rangle).$

$p1 = \text{person}(\text{Name}, \text{Surname}).$

При согласовании в первый раз, переменная $p1$ свяжется с $\text{person}(\langle \text{Anton} \rangle, \langle \text{Timonin} \rangle).$ Во второй раз, так как переменные Name и Surname сопоставимы с $\langle \text{Anton} \rangle, \langle \text{Timonin} \rangle,$ и тогда в переменная Name свяжется с $\langle \text{Anton} \rangle,$ а переменная Surname свяжется с $\langle \text{Timonin} \rangle$

Какое первое состояние резольвенты?

— Вопрос

Как применяется подстановка, полученная с помощью алгоритма унификации?

— Если унификация прошла успешно, то применяется подстановка. Переменные связываются со значениями

Как формируется новое состояние резольвенты?

— Преобразование резольвенты выполняется с помощью редукции. Редукция замена цели телом того правила, заголовок которого унифицируется с целью. Новая резольвента получается в два этапа:

1. В текущей резольвенте выбирается одна из целей и для неё выполняется редукция, затем получаем новую резольвенту
2. К новой, полученной резольвенте применяется подстановка, как наибольший общий унификатор цели и заголовка правила, сопоставимого с этой целью.

Как выделить за один шаг первые два подряд идущих элемента списка?

$p1 = [1, 2, 3, 4, 5].$

$p1 = [A \mid [B \mid C]].$

Тогда $A = 1, B = 2$.

Как выделить 1-й и 3-й элемент за один шаг?

$p1 = [1, 2, 3, 4, 5]$.

$p1 = [A \mid _ \mid [B \mid C]]]$.

Тогда $A = 1, B = 3$.

Исправления 2

№20

Как организуется хвостовая рекурсия в Prolog?

— Рекурсия в Prolog организуется следующим образом:

- рекурсивный вызов должен быть ПОСЛЕДНЕЙ подцелью;
- нужно избавиться от точек возврата с помощью отсечения (предикат "!"), чтобы исключить возможные альтернативы.

Какое первое состояние резольвенты?

— Первое состояние резольвенты - вопрос

Каким способом можно разделить список на части, какие, требования к частям?

— В Prolog для деления списка на части есть специальный символ «|». Этот символ разделяет список на голову и хвост. Также этим

специальным символом можно разделять произвольное число начальных элементов списка.

Как выделить за один шаг первые два подряд идущих элемента списка? Как выделить 1-й и 3-й элемент за один шаг?

— $[A, B \mid _]$ - A - первый элемент списка, B - второй элемент списка

— $[A, _, B \mid _]$ - A - первый элемент списка, B - третий элемент списка

Как применяется подстановка, полученная с помощью алгоритма унификации?

— Подстановка, полученная с помощью алгоритма унификации применяется следующим образом: в подстановке $\{X_1=T_1, \dots, X_n=T_n\}$ каждое вхождение переменной X_i заменяется на соответствующий терм T_i .