|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Отчет**

**по лабораторной работе № 8**

|  |  |
| --- | --- |
| **Дисциплина:** Функциональное и логическое программирование  **Студент:** Платонова Ольга  **Группа:** ИУ7-65Б  **Преподаватели:** Толпинская Н. Б.  Строганов Ю. В. |  |

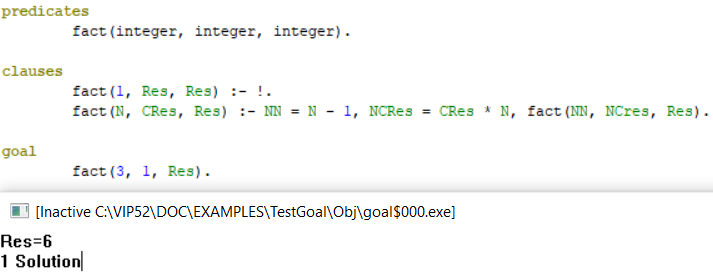
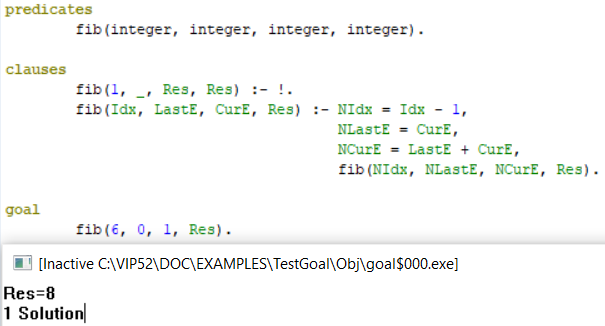
Москва, 2021 г.

***Задание***

Используя хвостовую рекурсию, разработать программу, позволяющую найти

1. n!
2. n-е число Фибоначчи.

***Листинг***

**1. fact(3, 1, Res).**

|  |  |  |  |
| --- | --- | --- | --- |
| № шага | Состояние резольвенты и вывод | Для каких термов запускается алгоритм унификации и каков результат | Дальнейшие действия |
| 0 | fact(3, 1, Res). |  |  |
| 1 | fact(3, 1, Res). | T1 = fact(3, 1, Res).  T2 = fact(1, Res, Res).  Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ |
| 2 | fact(3, 1, Res). | T1 = fact(3, 1, Res).  T2 = fact(N, CRes, Res).  Успех. Унифицируемые.  Подстановка:  {N = 3, CRes = 1, Res = Res} | Удаляется из стека:  fact(3, 1, Res).  Связываются переменные:  N = 3, Cres = 1 |
| 3 | NN = 3 - 1,  NCRes = 1 \* 3,  fact(NN, NCres, Res). | NN = 3 – 1.  NN = 2 | Удаляется из стека:  NN = 3 – 1.  Связываются переменные:  NN = 2 |
| 4 | NCRes = 1 \* 3,  fact(2, NCres, Res). | NCRes = 1 \* 3.  NCRes = 3 | Удаляется из стека:  NCRes = 1 \* 3.  Связываются переменные:  NCRes = 3 |
| 5 | fact(2, 3, Res). | T1 = fact(2, 3, Res).  T2 = fact(1, Res, Res).  Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ |
| 6 | fact(2, 3, Res). | T1 = fact(2, 3, Res).  T2 = fact(N, CRes, Res).  Успех. Унифицируемые.  Подстановка:  {N = 2, CRes = 3, Res = Res} | Удаляется из стека:  fact(2, 3, Res).  Связываются переменные:  N1 = 2, CRes1 = 3 |
| 7 | NN1 = 2 - 1,  NCRes1 = 3 \* 2,  fact(NN1, NCres1, Res1). | NN1 = 2 – 1.  NN1 = 1 | Удаляется из стека:  NN1 = 2 – 1.  Связываются переменные:  NN1 = 1 |
| 8 | NCRes1 = 3 \* 2,  fact(1, NCres1, Res1). | NCRes1 = 3 \* 2.  NCRes1 = 6 | Удаляется из стека:  NCRes1 = 3 \* 2.  Связываются переменные:  NCRes1 = 6 |
| 9 | fact(1, 6, Res1). | T1 = fact(1, 6, Res1).  T2 = fact(1, Res, Res).  Успех. Унифицируемые.  Подстановка:  {1 = 1, Res = 6, Res = Res1 } | Удаляется из стека:  fact(1, 6, Res).  Связываются переменные:  Res = 6, Res1 = 6 |
| 10 | !. | !.  Истина. | Удаляется из стека:  !. |
| 11 | Резольвента пуста. |  | Выводится Res = 6  Развязываются переменные:  Res, Res1  Откат. |
| 12 | !. | !.  Завершение процедуры. | Удаляется из стека:  !.  Развязываются переменные:  NCRes1, NN1, N1, Cres1, NCRes, NN, N, CRes |
| 13 | Резольвента пуста. |  | Завершение работы программы. |

**2. fib(4, 0, 1, Res).**

|  |  |  |  |
| --- | --- | --- | --- |
| № шага | Состояние резольвенты и вывод | Для каких термов запускается алгоритм унификации и каков результат | Дальнейшие действия |
| 0 | fib(4, 0, 1, Res). |  |  |
| 1 | fib(4, 0, 1, Res). | T1 = fib(4, 0, 1, Res).  T2 = fib(1, \_, Res, Res).  Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ |
| 2 | fib(4, 0, 1, Res). | T1 = fib(4, 0, 1, Res).  T2 = fib(Idx, LastE, CurE, Res).  Успех. Унифицируемые.  Подстановка:  {Idx = 4, LastE = 0, CurE = 1, Res = Res} | Удаляется из стека:  fib(4, 0, 1, Res).  Связываются переменные:  Idx = 4, LastE = 0, CurE = 1 |
| 3 | NIdx = 4 - 1,  NLastE = 1,  NCurE = 0 + 1,  fib(NIdx, NLastE, NCurE, Res). | NIdx = 4 – 1.  NIdx = 3 | Удаляется из стека:  NIdx = 4 – 1.  Связываются переменные:  NIdx = 3 |
| 4 | NLastE = 1,  NCurE = 0 + 1,  fib(3, NLastE, NCurE, Res). | NLastE = 1 | Удаляется из стека:  NLastE = 1.  Связываются переменные:  NLastE = 1 |
| 5 | NCurE = 0 + 1,  fib(3, 1, NCurE, Res). | NCurE = 0 + 1.  NCurE = 1 | Удаляется из стека:  NCurE = 0 + 1.  Связываются переменные:  NCurE = 1 |
| 6 | fib(3, 1, 1, Res). | T1 = fib(3, 1, 1, Res).  T2 = fib(1, \_, Res, Res).  Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ |
| 7 | fib(3, 1, 1, Res). | T1 = fib(3, 1, 1, Res).  T2 = fib(Idx1, LastE1, CurE1, Res1).  Успех. Унифицируемые.  Подстановка:  {Idx1 = 3, LastE1 = 1, CurE1 = 1, Res1 = Res} | Удаляется из стека:  fib(3, 1, 1, Res).  Связываются переменные:  Idx1 = 3, LastE1 = 1, CurE1 = 1 |
| 8 | NIdx1 = 3 - 1,  NLastE1 = 1,  NCurE1 = 1 + 1,  fib(NIdx1, NLastE1, NCurE1, Res11). | NIdx1 = 3 – 1.  NIdx1 = 2 | Удаляется из стека:  NIdx1 = 3 – 1.  Связываются переменные:  NIdx1 = 2 |
| 9 | NLastE1 = 1,  NCurE1 = 1 + 1,  fib(2, NLastE1, NCurE1, Res11). | NLastE1 = 1. | Удаляется из стека:  NLastE1 = 1.  Связываются переменные:  NLastE1 = 1 |
| 10 | NCurE1 = 1 + 1,  fib(2, 1, NCurE1, Res11). | NCurE1 = 1 + 1.  NCurE1 = 2. | Удаляется из стека:  NCurE1 = 1 + 1.  Связываются переменные:  NCurE1 = 2 |
| 11 | fib(2, 1, 2, Res11). | T1 = fib(2, 1, 2, Res11).  T2 = fib(1, \_, Res, Res).  Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ |
| 12 | fib(2, 1, 2, Res11). | T1 = fib(2, 1, 2, Res11).  T2 = fib(Idx2, LastE2, CurE2, Res2).  Успех. Унифицируемые.  Подстановка:  {Idx2 = 2, LastE2 = 1, CurE2 = 2, Res2 = Res11} | Удаляется из стека:  fib(2, 1, 2, Res11).  Связываются переменные:  Idx2 = 2, LastE2 = 1, CurE2 = 2 |
| 13 | NIdx2 = 2 - 1,  NLastE2 = 2,  NCurE2 = 1 + 2,  fib(NIdx2, NLastE2, NCurE2, Res2). | NIdx2 = 2 – 1.  NIdx2 = 1 | Удаляется из стека:  NIdx2 = 2 – 1.  Связываются переменные:  NIdx2 = 1 |
| 14 | NLastE2 = 2,  NCurE2 = 1 + 2,  fib(1, NLastE2, NCurE2, Res2). | NLastE2 = 2. | Удаляется из стека:  NLastE2 = 2.  Связываются переменные:  NLastE2 = 2 |
| 15 | NCurE2 = 1 + 2,  fib(1, 2, NCurE2, Res2). | NCurE2 = 1 + 2.  NCurE2 = 3 | Удаляется из стека:  NCurE2 = 1 + 2.  Связываются переменные:  NCurE2 = 3 |
| 16 | fib(1, 2, 3, Res2). | T1 = fib(1, 2, 3, Res2).  T2 = fib(1, \_, Res3, Res3).  Успех. Унифицируемые.  Подстановка:  {1 = 1, \_ = 2, Res3 = 3, Res3 = Res2} | Удаляется из стека:  fib(1, 2, 3, Res2).  Связываются переменные:  Res3 = 3, Res2 = 2 |
| 17 | !. | !.  Истина. | Удаляется из стека:  !. |
| 18 | Резольвента пуста. |  | Выводится Res = 3  Развязываются переменные:  Res2, Res3  Откат. |
| 19 | !. | !.  Завершение процедуры. | Удаляется из стека:  !.  Развязываются переменные:  NCurE2, NLastE2, NIdx2, Idx2, LastE2, CurE2, NCurE1, NLastE1, NIdx1, Idx1, LastE1, CurE1, NCurE, NLastE, NIdx, Idx, LastE, CurE |
| 20 | Резольвента пуста. |  | Завершение работы программы. |

***Вывод***

Эффективность работы системы может быть достигнута за счет хвостовой рекурсии и использования отсечения (уменьшения количества унификаций) в тех случаях, когда заведомо известна единственность ответа на вопрос.

1. *Что такое рекурсия? Как организуется хвостовая рекурсия в Prolog? Как организовать выход из рекурсии в Prolog?*

Рекурсия – определение объекта через ссылку на самого себя. Один из способов организации повторных вычислений. Для организации хвостовой рекурсии необходимо, чтобы рекурсивный вызов был последним в теле рекурсивного правила, и не оставалось других точек выбора. Выход из рекурсии осуществляется либо достижением базиса рекурсии, либо условием в теле правила.

1. *Какое первое состояние резольвенты?*

Исходная резольвента содержит вопрос.

1. *В каком случае система запускает алгоритм унификации? Каково назначение использования алгоритма унификации? Каков результат работы алгоритма унификации?*

Система запускает алгоритм унификации, когда резольвента не пуста. Алгоритм унификации необходим для того, чтобы подобрать знание, чтобы ответить на поставленный вопрос. Результатом работы алгоритма является значение переменной «неудача». Если неудача = 1, то унификация невозможна; если неудача = 0, то унификация прошла успешно, а побочным действием работы алгоритма является содержимое результирующей ячейки – результирующая подстановка.

1. *В каких пределах программы уникальны переменные?*

Именованные переменные уникальны в рамках предложения, анонимные – любые уникальны.

1. *Как применяется подстановка, полученная с помощью алгоритма унификации?*

В результате подстановки связываются переменные, которые еще не были связаны. После связывания всех утверждений, будет напечатано значение связанных переменных.

1. *Как изменяется резольвента?*

Резольвента меняется в 2 этапа:

* 1. Редукция (замена вопроса на тело правила, заголовок которого был успешно унифицирован);
  2. Применение подстановки.

1. *В каких случаях запускается механизм отката?*

В случае, когда унификация на текущем шаге завершается тупиковой ситуацией, или был получен ответ «да».