

Практическое занятие «Пролог-3»

24 сентября 2019 года

1. Без использования стандартных предикатов для работы со списками (**reverse**, **prefix**, **subffix**, **append** и т.д.) напишите предикат, принимающий два списка и истинный, если второй список есть обращение первого. С его использованием напишите предикат **palindrome/1(+list)**, принимающий список, не включающий элементов, кроме атомов 'a', 'b', 'c', 'A', 'B', 'C', и истинный, если этот список является палиндромом (одинаково читается в обоих направлениях) без учета регистра символов атомов. Например,
`palindrome(['a', 'c', 'c', 'A']) → true`,
`palindrome(['a', 'c', 'B']) → false`.
2. Напишите предикат, принимающий два списка и истинный, если второй список получается из первого выкидыванием составных чисел. Порядок оставшихся элементов должен сохраниться. Используйте предикат проверки числа на простоту, написанный на прошлом занятии.
3. Не используя стандартные предикаты работы со списками (**append**, **prefix** и т.д.), напишите предикат, принимающий два списка и истинный, если первый список есть начало второго. С использованием созданного предиката, напишите предикат, принимающий два списка и истинный, если первый список есть (непрерывный) подсписок во второго. Проверьте работу ваших предикатов в режиме проверки и в различных режимах поиска.
4. Не используя стандартные предикаты работы со списками (**append**, **sort** и т.д.), напишите предикат, принимающий три числовых списка, отсортированных по возрастанию, и истинный, если третий список есть слияние первых двух (то есть, содержит элементы первых двух в соответствующем количестве и соответствующем порядке).
5. Напишите предикат **numlst/2(N,Lst)**, истинный, если **Lst** есть список цифр неотрицательного целого числа **N**, перечисленных от старших к младшим разрядам.
 - а) Разработайте его вариант, который работает в виде **numlst/2(+N,?Lst)**;
 - б) Расширьте уже сделанный вариант предиката, чтобы он работал и в виде **numlst/2(?N,?Lst)**; придумайте разумное поведение предиката для случая, когда оба аргумента свободны.
- 6*. Напишите собственную реализацию сортировки Хоара (быстрой сортировки) заданного списка (предикат, принимающий два списка и истинный, если второй есть отсортированная версия второго).
- 7*. Напишите предикат, принимающий два списка и истинный, если второй список есть максимальный по длине непрерывный возрастающий подсписок первого списка.