Task B

The predicate takes a list of persons, a minimum height and an age range. The base case when the list is empty, the predicate returns “No match found” meaning no match was found. If the list isn’t empty, it checks if the first persons height is taller than the inputted height. If it’s true it continues and calls the helper predicate inputting the original age range as well as the persons age. If this is also true, the (!) cut operator stops the backtracking. If it reaches the last person in the list this special case will require a separate predicate to allow the code to continue as it should.

A screenshot of a computer

Description automatically generated

Task C

The goal is to find the first 5 prime numbers and filtering out the rest, then reverses the digits of each one. Filter\_and\_transform takes an input and calls the helper with the input, an empty, list, result and a 0 value for the counter. If the input is empty, and empty list is returned. If the counter value is 5 it calls the reverse predicate on the list and then cuts the backtracking once it is returned. Otherwise if neither of these conditions are met, it checks if the head of the inputted list is a prime number using is\_prime, if it is a prime number, it is reversed using reverse\_number. The count is incremented and the new number added to the accumulator list. While this process repeats on the tail of the list. Is\_prime checks if the number is greater than 1 first, then does divisibility tests on it using the helper. Is\_prime\_helper checks if the square of the current divisor is greater than the number as that would mean the number is prime. The other helper checks the number is not divisible by the current divisor, and repeats until the first helper clause is met.

A screenshot of a computer

Description automatically generated