Question 1:

(a) Let \( f(x) = 1 \) if \( x \) is even, \( f(x) = 2 \) if \( x \) is odd. Show that \( f(x) \) is primitive recursive.

(b) Show that \( f(x) \) is partially computable by writing a program in language \( L \) that computes \( f(x) \). Do not use any macros but only the proper \( L \) commands. By the way, this will allow you to test your program using the Haskell code.

Question 2:

Let \( g(x) = 2x \) if \( x \) is a perfect square, \( g(x) = 2x + 1 \) otherwise. Show that \( g(x) \) is primitive recursive.

Question 3:

Let \( h(x) \) be the number of primes that are less than \( x \). Show that \( h(x) \) is primitive recursive.

Question 4:

Let \( k(x) \) be the integer \( n \) such that \( n \leq \sqrt{2x} \leq n + 1 \). Show that \( k(x) \) is primitive recursive.