1. Define a function `lsf` which takes an integer `n` as input and returns the largest integer whose square divides `n`.
   You may define auxiliary functions.

```
- lsf;
val it = fn: int -> int
- lsf(5);
val it = 1: int
- lsf(12);
val it = 2: int
- lsf(15);
val it = 1: int
- lsf(28);
val it = 2: int
- lsf(100);
val it = 10: int
```

2. Fibonacci numbers are defined as follows: \( f_0 = 1 \), \( f_1 = 2 \) and \( f_n = f_{n-1} + f_{n-2} \) for all \( n > 1 \). Define a function `sum_of_fib` which takes an integer `n` as input and checks whether it is the sum of two Fibonacci numbers.
   You may define auxiliary functions.

```
- sum_of_fib;
val it = fn: int -> bool
- sum_of_fib(7);
val it = true: bool
- sum_of_fib(12);
val it = false: bool
- sum_of_fib(13);
val it = true: bool
- sum_of_fib(14);
val it = true: bool
- sum_of_fib(17);
val it = false: bool
- sum_of_fib(433494438);
val it = true : bool
```