1 The problem

This is a problem 70 from Project Euler.

2 Definitions

This problems deals with $\varphi(n)$ function. This function calculation is based on prime divisors of its argument:

$$\varphi(n) = \prod_{i=1}^{k} p_i^{\alpha_i - 1} (p_i - 1)$$

Or, if we don't want to care about primes' powers:

$$\varphi(n) = n \prod_{p|n} \left(1 - \frac{1}{p}\right)$$

$\varphi(n)$ is already implemented so let's try just to reuse it from Math.Sieve.Phi (just as not cool as implementing by hand):

```
import Math.Sieve.Phi (sieve, phi)
import Data.List (sort, sortBy)
import Data.Ord (comparing)


is_permutation :: (Show a) ⇒ a → a → Bool
is_permutation a b = (sort (show a)) ≡ (sort (show b))


max_n = 10000000


permuts :: [(Int, Int)]
permuts = sortBy (comparing (λ(n, p) → (fromIntegral n) / (fromIntegral p))) $ filter (λ(n, p) → is_p
  where si = sieve max_n


solve :: Int → [(Int, Int)]
solve a = take a permuts


main :: IO ()
main = print $ solve 1
```

That's pretty fast (and dirty), but not compiles - works only in ghci. Not that I care.