Introduction to Computer Science
Jacobs University Bremen
Dr. Jürgen Schönwälder

Course: CH-232-A
Date: 2019-09-27
Due: 2019-10-04

ICS 2019 Problem Sheet #3

Problem 3.1: *size of power sets (proof by induction)* (3 points)

Prove the following statement by induction:

The number of elements in the power set $\mathcal{P}(S)$ of a finite set S with n elements is 2^n.

Problem 3.2: *reflexive, symmetric, transitive* (1+1+1 = 3 points)

For each of the following relations, determine whether they are reflexive, symmetric, or transitive. Provide a reasoning.

a) $R = \{(a,b) | a,b \in \mathbb{Z} \wedge a \neq b\}$
(The numbers a and b are different.)

b) $R = \{(a,b) | a,b \in \mathbb{Z} \wedge |a-b| \leq 3\}$
(The absolute difference of the numbers a and b is less than or equal to 3.)

c) $R = \{(a,b) | a,b \in \mathbb{Z} \wedge (a \bmod 10) = (b \bmod 10)\}$
(The last digit of the decimal representation of the numbers a and b is the same.)

Problem 3.3: *circular prime numbers (haskell)* (2+2 = 4 points)

A *circular prime* is a prime number with the property that all numbers generated by cyclically permuting its (base 10) digits will be prime. For example, 1193 is a circular prime number since 1193, 1931, 9311, and 3119 are all prime numbers. Some example usages:

a) Implement a function `isPrime :: Integer -> Bool` indicating whether the argument is a prime number or not.

```
> isPrime 2
True
> filter isPrime [2..100]
[2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,97]
```

Explain how your function works and how you have tested it.

b) Using the `prime` function, implement a function `isCircPrime :: Integer -> Bool` indicating whether the argument is a circular prime number or not.

```
> filter isCircPrime [2..100]
[2,3,5,7,11,13,17,31,37,71,73,79,97]
```

Explain how your function works and how you have tested it.

Submit your Haskell source code as a plain text file.

Hints:

- The Haskell `div` function returns how many times the first number can be divided by the second one and the `mod` function returns the remainder after division of the first number by the second.
- You can reuse the `rotate` and `circle` functions from the last assignment to solve this problem. To convert an Integer value into a string of (base 10) digits, you can use the `show :: Show a => a -> String` function. To convert a string of (base 10) digits into an Integer, you can use the `read :: Read a => String -> a` function. Note that the type must be clear from the context. If now, you need to tell `read` to which type a string should be converted.

```
> show 42
"42"
> read "42" :: Integer
42
```