Problem Sheet #2

Name: Flori Kusari

Problem 2.1: Proof by Contrapositive

Prove: "If a^{32} is an odd number, then a^4 is an odd number".

Contrapositive: If a^4 is not an odd number, then a^{32} is not an odd number.

Assuming a^4 is not an odd number means it is an even number. This even number can be expressed as "$2k$" for "k" being an integer.

Now to prove we express the following:

$$(a^4)^8 = (2k)^8$$

$$a^{32} = (2k)^8$$

$$a^{32} = (2)^8 (k)^8$$

$$a^{32} = 256(k)^8$$

Since 256 is an even number we can do what we did before and write it as 2 times some integer "x".

$$a^{32} = 2x(k)^8$$

Expressing a^{32} as $2 \cdot x$ times some integer proves it is an even number. Thus, a^4 is not an odd number and a^{32} is not an odd number which is the contrapositive of the original statement.

By proving it's contrapositive we have also proven that the original statement is true. □

BRUNNEN

Problem 2.2: Prove by induction

Prove: $n^3+(n+1)^3+(n+2)^3$ is divisible by 9
For $n \in N$ and $n \geq 1$

Base Case: For $n=1$

$$(1)^3+(1+1)^3+(1+2)^3 = 36 = 9 \cdot 4$$

The expression in the base case is divisible by 9.

Assume that since $(\forall n \in N)$ and $(n \geq 1)$, the expression is divisible by 9

$$n^3+(n+1)^3+(n+2)^3 = 9x$$

for some "x" integer.

Test Case: Prove the statement for $(n+1)$:

$$\underbrace{n^3+(n+1)^3+(n+2)^3}_{\text{divisible by } 9} + (n+3)^3 - n^3$$

$= 9x$ for "x" a positive integer.

$$\underbrace{n^3+(n+1)^3+(n+2)^3}_{=9x} + \cancel{n^3} + \underbrace{9n^2}_{9x} + \underbrace{27n}_{9\cdot 3n} + \underbrace{27}_{9\cdot 3} - \cancel{n^3}$$

Expression is divisible by 9

Everything is divisible by 9.

From the Base Case and the Test Case, we can conclude that $n^3+(n+1)^3+(n+2)^3$ is divisible by 9 for $n \in N$ and $n \geq 1$ thus the statement has been proved through induction.

Problem 2.3:

a)

```
divisors :: Int -> [Int]
divisors n = [d | d <- [1..n], n `mod` d == 0]
```

It iterates from 1 to "n" using "d" to store the current iteration.

Creates a list of integers from 1 to n

Checks if the remainder is 0 in the division of "n" and "d" in which case it is counted as a divisor

The expression is in [ ] because the answer will be a list that holds all divisors of "n".

b)

calculates the "d" raised power of "z"

each divisor in the list

```
sigma :: Int -> Int -> Int
sigma z n = sum [d^z | d <- divisors n]
```

calculates the sum of all these powers "z".

generate a list of all "z"-th power.

calls the previous function

Function sigma uses 2 arguements "z" and "n".

The result is the sum of the "z"-th powers of the divizors of "n" taken from the previous function.

BRUNNEN