Exercise 3

# Exercise 3.1: Patterns

<pattern> ::= <name>

| <constant>

| “\_”

| <pattern> :: <pattern>

| ( <pattern> , <pattern> )

| [ <pattern> {, <pattern>} ]

# 

# Exercise 3.2: Polymorphism

1. i + r = real
   1. i+r -> r
2. i + r + i = real
   1. i+r -> r
   2. r+i -> r
3. i + (r + i) = real
   1. (r+i) -> r
   2. i+r -> r
4. i + i + r + (r + i) = real
   1. (r+i) -> r
   2. i+i -> i
   3. i+r -> r
   4. r+r -> r

# Exercise 3.3: Types

fun maxpairs ((x, y) :: xs) = (if x > y then x else y) :: maxpairs xs

The function first checks if there is an empty list and returns an empty list. SOSML doesnt want to run without this safety feature.

If there are tuples in the list then the function will take the first set and compare it with an if then else, then recursively call the function to do it again with the next set of tuples in the list.