Solace Running Documentation

Noé Garcia

March, 2023

1 Introduction

This document is a running documentation for the Solace language. This documentation contains information regarding value types, and general syntax along with simple examples.

2 Types

Solace is a statically typed language. All values are also immutable. The following table contains all available types in Solace:

Solace Type	Description
int	general integer
float	general floating point number
char	general character
string	general string value
bool	general boolean value
:sym	symbol value
func	function type

2.1 Integers and Floats

Built in number values are represented within Solace as integers or floats. Much like other languages, integers represent whole values, and floats represent floating point real values. integer values can be defined in the following manner:

```
module:Main

func main int()
{
        // Declare a to be an integer and b to be a float
        a int = 1;
        b float = 0.1;

        // values of the same type can be defined with the use of commas:
        c, d int = 2, 3;

        0;
}
```

There are a number of arithmetic operations that are available for both integers and floats. Like other languages, number values are able to be added, subtracted, multiplied, and divided. Numbers are also able to be compared between one another. The following table contains operators for number types in Solace:

Operators	Description
+	addition
−	subtraction
*	multiplication
/	division
=	value assignment
==	equal to
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
!=	not equal to

Arithmetic and comparison operators are performed between two given values. The following is an example of each operator usage:

```
module : Main

fun main int ()
{
        // assume a and b are integers:
        a, b int = 5, 10;

        // arithmetic can be performed
        addition int = a + b;
        subtraction int = b - a;
        division int = a / b;
        multiplication int = a * b;

        // numbers can also be compared:
        a > b;  // false
        a < b;  // true
        a >= b; // false
        a <= b; // true
        a == b; // false
        a != b; // true

        // The above can also be done with float type variables, or between integers
        // and float variables.

        0;
}
```

2.2 Strings and Characters

things

2.3 Boleans and Symbols

things

2.4 functions

things

3 Complex Data Types

things

4 Controll Flow

things