Quick Reference Manual

Welcome to the QRM for YACCP programming language, here you'll find all you need to get you started.

How does the project work?

The project is divided basically into two parts in which different classes divide the work to generate an output for the code that is written, we have the compilation process and the execution process, the first process reads the input file and identifies by comparing to the lexer and rules in the parser what the instructions will be, then also during this process the QuadrupleGen and QuadActions classes come in play. They, with the help of other structures such as the Symbol Table and Function Directory, will translate the code via Intermediate Code Generation by generating Quadruples, this quadruples will be then passed on to the next process in which the Virtual Machine with the help of the memory and the function directory will execute and throw the output for the file.

How can I write a program for this language?

The easiest way to do it is to go to the rules and check how the structure of different statements and declarations are built, although you can also use this quick guide to how a program is built.

A program consists of the following elements:

Program <program name>

For example: `program newprogram`

The following are optional up until the main function

Var declaration (int, float and char are the only types supported at the moment)

For example:

- `int variable1;`
- `float variable2;`

Function declarations look like this:

```
Func <return type> name(type var, type var, …){


}
```

You can declare variables at the beginning of a function just like this:

```
Func int function1(int param1, float param2){

      int variable1;
      float variable2;



}
```

Finally after declaring both variables and functions (or not, since they're optional) you have to declare the main, this will be the function that runs the code you provide, consider it as the execution function. To declare main you simply write the following

```
main(){


}
```

* It's important to note that variable declaration is not available inside of main so all variables must be declared at the beginning of the program

Other statements you can use are the while loop and the if-else condition

The while loop looks like this:

```
while(condition){


}
```

The if-else condition looks like this:

```
if(condition){

}

else {


}
```

Finally two more tools to get you started, print and read, print will show in the console whatever you put inside it, meanwhile read, will read a value and assign it to the variable you are reading, they look like this:

```
read(variable2);
print(variable1);
```

If you combine all the snippets you'll end up with a program like this:

```
program newprogram
int variable1;
float variable2;

Func int function1(int param1, float param2){
	int localvar1;
	float localvar2;

	if(localvar1 > variable1){
		variable = localvar1 + 1;
	}
	else {
		variable2 = localvar2 + 2.0;
	}

}
main(){
	read(variable2);
	variable1 = function1(variable1, variable2);
	print(variable1);
}
```

Hope this was helpful for your start, have fun and if you have other questions on how to use certain feature, you can always refer to the documentation.