1. Run `make` to build the code generator.

2. Run `java -cp deps:. P6 <Carrot-Source-File> <MIPS-Output-File>` to generate MIPS assembly of a Carrot source file. I use the following code, which consists of three ways to calculate the fibonacci number,

```
1   int a0;
2   int a1;
3
4   int fib1(int n) {
5       if (n == 0) {
6           return 0;
7       }
8
9       if (n < 2) {
10          int t;
11          t = a1;
12          return t;
13      } else {
14          return fib1(n - 1) + fib1(n - 2);
15      }
16  }
17
18  int fib2(int n, int a, int b) {
19      if (n < 1) {
20          return a;
21      } else {
22          int r;
23          int t;
24          t = a + b;
25          r = fib2(n - 1, b, t);
26          return r;
27      }
28  }
29
30  int fib3(int n) {
31      int a;
32      int b;
33      a = a0;
34      b = 1;
35      while (n > 0) {
36          int t;
37          n--;
38          t = a + b;
39          a = b;
40          b = t;
41      }
42      return a;
43  }
44
```
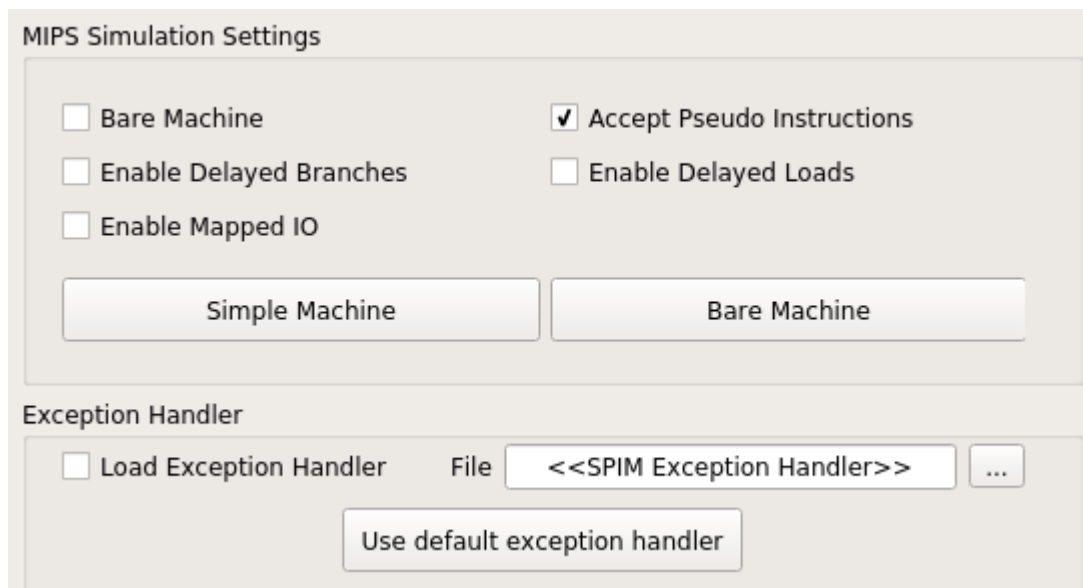
```
45   void main() {
46       a0 = 0;
47       a1 = 1;
48       cout << fib1(10);
49       cout << "\n";
50       cout << fib2(10, 0, a1);
51       cout << "\n";
52       cout << fib3(10);
53   }
```

and run `java -cp deps:. P6 fib.crrt fib.s`, then a MIPS assembly file `fib.s` is generated.

3. Run QtSpim and make its settings as following in order to run the generated assembly successfully,

MIPS Simulation Settings

☐ Bare Machine   ✔ Accept Pseudo Instructions
☐ Enable Delayed Branches   ☐ Enable Delayed Loads
☐ Enable Mapped IO

[Simple Machine   ]   [Bare Machine   ]

Exception Handler

☐ Load Exception Handler   File [<<SPIM Exception Handler>>] [...]

[Use default exception handler]

4. Load the `fib.s` file into QtSpim and press F5 to run the code. The QtSpim console will print three 55's, just as we have written in the Carrot code.

5. To clean up the class files, run `make clean`.