Jack Dong
CSCI3081W
Refactoring 1 - revision

Refactoring 1: Extract Method

src/config_sim_helper.h

- method CheckFile() was created - line 83
- method GetBusTimeLength() was created - line 104
- method GetSimLength() was created - line 45
- method GetYesNoAns(bool) was created - line 56
    - Used in methods Arg1(), Arg2(), Arg3(), Arg 4() - lines 115, 130, 133, 160
        - Used in method ArgumentHandling() - line 175
            - Used in main - line 201

Branch 16-config-sim accounts for only 2 args and was initially in drivers/configuration_sim.cc. Each branch after 16 until 25 adds to the program.

Since my goals were to implement parameter handling for different number of arguments, I had many repeat functions such as (1) check if simulation time was a positive number (2) check if bus deployment time is a positive number (3) check if inputted file can be opened. Since I wanted my program to handle one to four arguments, these three functions had to be applied four times. In addition, I extracted how the program handles each number of arguments from the main method so (into ArgN(...) functions where N is an integer from one to four). This way parameter handling goes from over half the main program to just one line. This structure makes the main function much cleaner, shorter, and it prevents readers from being overwhelmed / distracted from the other functionalities of main().

```
/* example helper method for parameter checking */
void CheckFile(char * filename) {
  const char * buffer = filename;
  std::strcpy(file_name_c, "./config/");
  std::strcat(file_name_c, buffer);
  in_file.open(file_name_c);

  if (!in_file) {
    std::cout << "Cannot open file: " << filename << std::endl;
    std::cout << "Default to config/config.txt? (y/n)" << std::endl;
    GetYesNoAns(0);
    file_name = "config.txt";
  } else {
    file_name = filename;
  }
  free(file_name_c);
  in_file.close();
}

/* example argument method */
void Arg4(char * filename, int SimLength, int BusLength) {
  CheckFile(filename);
  sim_length = SimLength;
  GetSimLength();
  bus_length = BusLength;
  GetBusTimeLength();
}

/* argument handling method */
void ArgumentHandling(int argc, char** argv) {
  if (argc == 1) {
    std::cout << "No file specified." << std::endl;
    Arg1();
  } else if (argc == 2) {
    Arg2(argv[1]);
  } else if (argc == 3) {
    Arg3(argv[1], atoi(argv[2]));
  } else if (argc == 4) {
    Arg4(argv[1], atoi(argv[2]), atoi(argv[3]));
  } else {
    std::cout << "Please follow usage guidelines." << std::endl;
    std::cout << "Usage: ./build/bin/config_sim <config_filename> " <<
    "<simulation_length> <time_between_busses>" << std::endl;
    exit(1);
  }
}

/* how main handles arguments */
int main(int argc, char**argv) {
                ...
                ArgumentHandling(argc, argv);
                …
}
```