

Programming Assignment 1: Benchmark Translator - Report

李品賢, 108501023, EE4B

(1) How to compile and execute your program; (You can use screenshot to explain)

☒ main function

In the main function (Fig. 1), I use argc and argv as input arguments on the command line. Then, I create the object myFile to read and write files, and I also create the object BT from the class Benchmark_translator to use the read_file() & print_result(). I will explain later.

```
38  int main(int argc, char *argv[]) {
39
40      fstream myFile;
41      Benchmark_translator BT;
42
43      myFile.open(argv[1], ios::in);
44      BT.read_file(myFile);
45      myFile.close();
46
47      myFile.open(argv[2], ios::out);
48      BT.write_file(myFile);
49      myFile.close();
50
51      return 0;
52 }
```

Fig. 1

☒ class Benchmark_translator

In the class Benchmark_translator (Fig. 2), here are 2 functions within the public access modifier to interact with the user (main function) so the user can read and write files. Also, here are 7 functions within the private access modifier with different functions. I will explain the detail later.

```

16     class Benchmark_translator {
17     public:
18         Benchmark_translator();
19         void read_file(fstream &);
20         void write_file(fstream &);
21
22     private:
23         string line_, func_, name_;
24         vector<string> input_, output_;
25         int inputCount_, outputCount_, gateCount_, typeCount_[8];
26         vector<Gate_info> gates_;
27
28         void delete_blank();
29         void func_judgement();
30         void process_txt();
31         void get_name();
32         void get_IO(vector<string> &, int *);
33         void get_logic_gate();
34         int find_integer();
35
36     };

```

Fig. 2

➤ **Variables brief explanation:**

- A. string line_
store the received input file by object myFile and getline function
- B. string func_
store the current state (module, input, output, and, nor...)
- C. string name_
store the file name (c17, c432, c880...)
- D. vector<string> input_
store all the input ports
- E. vector<string> output_
store all the output ports
- F. int inputCount
store the input port count
- G. int outputCount
store the output port count
- H. int gateCount
store the overall logic gate count

- I. `int typeCount[8]`
store the different logic gate count (and, or, nand...)
- J. `vector<Gate_info> gates_` (Fig. 3)
store the information of gate including ports, port count and type of it

```

9      struct gate_info {
10         int port_cnt = 0;
11         vector<string> port;
12         string type = "none";
13     };
14     typedef struct gate_info Gate_info;

```

Fig. 3

➤ **Functions brief explanation:**

- A. `read_file()`
to read the file (.v) line by line, and also call other functions to process the txt.
- B. `write_file()`
write the file (.bench) with the fixed format.
- C. `delete_blank()`
delete all blanks within the received line string.
- D. `func_judgement()`
 - 1. function like a finite state machine.
 - 2. change state when receive the key word (module, input, and, nor...).
- E. `process_txt()`
 - 1. process the text depending on the current state.
 - 2. If the current state is "module", program will call `get_name()` to receive file name (c17, c432, c880...).
 - 3. If the current state is "input"/"output", program will call `get_IO()` to deal with the special format of I/O port.
 - 4. If the current state is concerned with logic gate, program will call `get_port()`, and also the corresponding `typeCount[]` will increase.
- F. `get_name ()`
capture the text between "module" and "(", which is file name.

G. `get_IO()` (Fig. 4)

1. If there exists ";" in string `line_`, replace the ";" with "," in order to make the following string processing more easily.
2. If "," still be found on the string `line_`, the while loop will keep processing, which means the here are remaining ports not recorded and need to be dealt with.
3. capture the text between 'index' storing the index of first integer in the string `line_` and ",", which is file name.
4. also, the `inputCount` / `outputCount` will increase by 1

```
185 void Benchmark_translator::get_IO(vector<string> &inout, int *inoutCount) {
186
187     // receive input/output ports in "input"/"output state
188
189     if(line_.rfind(";") != string::npos) {
190         line_ = line_.substr(0, line_.rfind(";")) + ",";
191         func_ = "none";
192     }
193
194     while(line_.find(",") != string::npos) {
195
196         (*inoutCount)++;
197
198         int index = find_integer();
199
200         inout.push_back(line_.substr(index, line_.find(",") - index));
201         line_ = line_.substr(line_.find(",") + 1, line_.length() - line_.find(",") - 1);
202     }
203 }
204 }
```

Fig. 4

H. `get_logic_gate()` (Fig. 5)

1. add "," in the end of string `line_in` in order to make the following string processing more easily.
2. This function is similar to `get_IO()`, but there are slight difference about the data structure in between.

```

208 void Benchmark_translator::get_logic_gate() {
209
210     // receive ports in logic gate state
211
212     line_ = line_.substr(line_.find("(") + 1, line_.find(")") - line_.find("(") - 1) + ",";
213
214     Gate_info temp;
215
216     gates_.push_back(temp);
217
218     while(line_.find(",") != string::npos) {
219
220         int index = find_integer();
221
222         gates_[gateCount_].port.push_back(line_.substr(index, line_.find(",") - index));
223         gates_[gateCount_].port_cnt++;
224
225         line_ = line_.substr(line_.find(",") + 1, line_.length() - line_.find(",") - 1);
226     }
227     gates_[gateCount_++].type = func_;
228 }
229 }

```

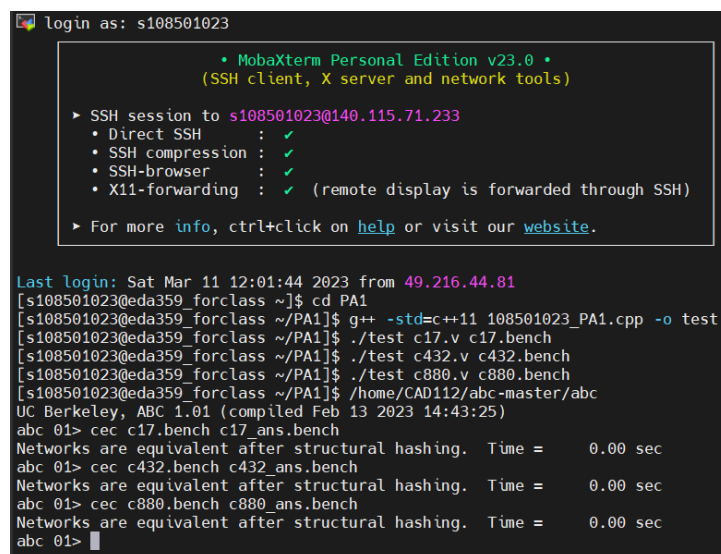
Fig. 5

I. find_integer()

started from left, search each character in string line_ and return the index of first integer.

(2) The completion of the assignment; (If you complete all requirements, just specify all)

According to the result by ABC below, I have done all the tasks. c17.bench, c432.bench and c880.bench are all equivalent with answers.



```

login as: s108501023

• MobaXterm Personal Edition v23.0 •
  (SSH client, X server and network tools)

► SSH session to s108501023@140.115.71.233
  • Direct SSH      : ✓
  • SSH compression : ✓
  • SSH-browser     : ✓
  • X11-forwarding  : ✓ (remote display is forwarded through SSH)

► For more info, ctrl+click on help or visit our website.

Last login: Sat Mar 11 12:01:44 2023 from 49.216.44.81
[s108501023@eda359_forclass ~]$ cd PA1
[s108501023@eda359_forclass ~/PA1]$ g++ -std=c++11 108501023_PA1.cpp -o test
[s108501023@eda359_forclass ~/PA1]$ ./test c17.v c17.bench
[s108501023@eda359_forclass ~/PA1]$ ./test c432.v c432.bench
[s108501023@eda359_forclass ~/PA1]$ ./test c880.v c880.bench
[s108501023@eda359_forclass ~/PA1]$ /home/CAD112/abc-master/abc
UC Berkeley, ABC 1.01 (compiled Feb 13 2023 14:43:25)
abc 01> cec c17.bench c17_ans.bench
Networks are equivalent after structural hashing. Time = 0.00 sec
abc 01> cec c432.bench c432_ans.bench
Networks are equivalent after structural hashing. Time = 0.00 sec
abc 01> cec c880.bench c880_ans.bench
Networks are equivalent after structural hashing. Time = 0.00 sec
abc 01>

```

Fig. 6

(3) The hardness of this assignment and how you overcome it;

Because I have taken the Data Structure course, I am familiar with C++ recently. But here are still some skills I didn't get used to them, such as the implement of class and vector. Fortunately, I picked up those skills after watching the online course of C++ programming produced by prof. Chen.

There were more tasks popping up in 2022/03/10, since I found big problem when I slight edited my code. I found that I used `getline()` in the IDE vs, the complier would not consider `"\n"` (ASCII 13) a character in string, while everything changed when I compiled in workstation... The compiler viewed `"\n"` as a character, so the error happened in my program when using `.length()`. There is one more character than I thought before. I spent 4 hours debugging just for this scenario. OuQ...

(4) Any suggestions about this programming assignment?

I think this project is a great material to practice C++ language. Like I said above, I picked up some missing skills and implemented it successfully. I am grateful for having this assignment.