

```
1: // $Id: matchlines.cpp,v 1.1 2016-06-28 14:46:02-07 - - $
2:
3: //
4: // Matches lines according to the program specs.
5: // Prints out results depending on type of line.
6: //
7:
8: #include <cassert>
9: #include <iostream>
10: #include <regex>
11: #include <string>
12: using namespace std;
13:
14: int main() {
15:     regex comment_regex {R"(\s*(#.*)?$)"};
16:     regex key_value_regex {R"(\s*(.?)\s*=\s*(.?)\s*$)"};
17:     regex trimmed_regex {R"(\s*([^\s=]+?)\s*$)"};
18:     for (;;) {
19:         string line;
20:         getline (cin, line);
21:         if (cin.eof()) break;
22:         cout << endl << "input: \"" << line << "\"" << endl;
23:         smatch result;
24:         if (regex_search (line, result, comment_regex)) {
25:             cout << "Comment or empty line." << endl;
26:             continue;
27:         }
28:         if (regex_search (line, result, key_value_regex)) {
29:             cout << "key : \"" << result[1] << "\"" << endl;
30:             cout << "value: \"" << result[2] << "\"" << endl;
31:         } else if (regex_search (line, result, trimmed_regex)) {
32:             cout << "query: \"" << result[1] << "\"" << endl;
33:         } else {
34:             assert (false and "This can not happen.");
35:         }
36:     }
37:     return 0;
38: }
39:
40: /*
41: //TEST// (echo '      # comment      ';
42: //TEST// echo '      ';
43: //TEST// echo '      No equal in this line.      ';
44: //TEST// echo '      key key key = value value value      ';
45: //TEST// echo 'only key here =      ';
46: //TEST// echo '      = only value here      ';
47: //TEST// echo '      =      ';
48: //TEST// ) | matchlines >matchlines.out 2>&1
49: //TEST// mkpspdf matchlines.ps matchlines.cpp* matchlines.out
50: */
51:
```

[illegible]

```
1:
2: input: "      # comment      "
3: Comment or empty line.
4:
5: input: "                                "
6: Comment or empty line.
7:
8: input: "      No equal in this line.          "
9: query: "No equal in this line."
10:
11: input: "      key key key  =  value value value  "
12: key   : "key key key"
13: value: "value value value"
14:
15: input: "only key here =          "
16: key   : "only key here"
17: value: ""
18:
19: input: "      = only value here          "
20: key   : ""
21: value: "only value here"
22:
23: input: "      =      "
24: key   : ""
25: value: ""
```