

```
1 /*
2  * Main.cpp
3  * Initiates the program
4  *
5  * Created: 1/3/2017 by Ryan Tedeschi
6  */
7
8 #include "../shared/Helpers/Helpers.h"
9 #include "ControlModule.h"
10 #include "Timestamp.h"
11 #include <iostream>
12 #include <string>
13 #include <list>
14
15 #define OTHER 0
16 #define MODULE 1
17 #define LANGUAGE 2
18 #define CODE 3
19 #define CODE_F 4
20 int SIZES[] { 0, 10, 12, 6, 7 };
21
22 int paramType(string);
23 string parseParam(string, int);
24 bool stripArgData(string, string*, string*);
25 string unescapeCharacters(string);
26
27 int main(int argCount, char** argArray)
28 {
29     cout << "Code Analyzer Software Package (CASP)" << endl;
30     cout << "Build Date: " << TIMESTAMP << endl;
31
32     vector<arg> fnArgs;
33     vector<string> codeSource;
34     string sourceLanguage;
35     string moduleID;
36
37     for (int i = 1; i < argCount; i++)
38     {
39         int type = paramType(argArray[i]);
40         string value = parseParam(argArray[i], type);
41         switch (type) {
42             case MODULE:
43                 moduleID = value;
44                 break;
45             case LANGUAGE:
46                 sourceLanguage = value;
47                 break;
48             case CODE_F:
49                 value = Helpers::ReadFile(value);
50             case CODE:
51                 codeSource.push_back(value);
52                 break;
53             default:
54                 string t, v;
55                 if (stripArgData(argArray[i], &t, &v)) {
56                     arg a(t, v);
57                     fnArgs.push_back(a);
58                 }
59         }
60     }
61
62     ControlModule control = ControlModule();
63     control.Run(sourceLanguage, moduleID, codeSource, fnArgs);
64
65     cout << "CASP - Operation Complete";
66 }
```

```
67     return 0;
68 }
69
70 int paramType(string input) {
71     int type = OTHER;
72     if (input.find("/sourcelang=") == 0)
73         type = LANGUAGE;
74     else if (input.find("/moduleid=") == 0)
75         type = MODULE;
76     else if (input.find("/code=") == 0)
77         type = CODE;
78     else if (input.find("/codef=") == 0)
79         type = CODE_F;
80     return type;
81 }
82
83 bool stripArgData(string input, string* id, string* value) {
84     int slash = input.find("/");
85     int eq = input.find("=");
86
87     if (slash != 0) {
88         *id = "";
89         *value = "";
90         return false;
91     }
92
93     if (eq == -1) {
94         *id = Helpers::toLower(input.substr(1, input.size()));
95         *value = "";
96         return true;
97     }
98
99     *id = Helpers::toLower(input.substr(1, eq - 1));
100     *value = input.substr(eq + 1, input.size());
101     return true;
102 }
103
104 string parseParam(string input, int type)
105 {
106     int size = SIZES[type];
107     string out = input.substr(size, input.size() - size);
108     return out;
109 }
```