

```
1 //Ben Scherer
2 // 7/7/2017
3 // Sinclair's got talent
4 // Judges enter scores for multiple contestant. Low/High score for each
   contestant dropped and average calculated. Highest average is declared
   winner.
5
6 //Headers to include
7 #include <iostream> //cout
8 #include <iomanip> // used to manipulate cout
9 #include <string> //needed for string variable
10 #include <math.h> //used for basic arithmetic
11 #include <limits> //user for numeric_limits
12 #include <utility> //needed for pair
13
14 using namespace std;
15
16 //global var
17 bool debug = false; //set to true to display debug output
18
19 //Functions Prototypes/Overrides
20 int getInput(string questionToAsk, string errorMsg, int lowRange, int
   highRange); //validates int input and int range
21 string getInput(string questionToAsk, string errorMsg); //validates string input
22 double calcAvgScore(int score1, int score2, int score3, int score4, int
   score5); //calculates avg score
23 int findLowest(int score1, int score2, int score3, int score4, int score5); //
   finds lowest int from 5 var
24 int findHighest(int score1, int score2, int score3, int score4, int score5); //
   finds highest int from 5 var
25 char getSentinel(); //Input validation for sentinel. simple y/n
26 pair<string, double> getScores();
27 bool checkInt(double num); //checks if number is a whole number
28
29
30 int main() {
31
32     pair<string, double>winner = getScores();//Get scores and find winner
33     cout << "\nThe winner of Sinclair's Got Talent! with an average score of " <<
   setprecision(2) << fixed << winner.second << " is ....." <<
   winner.first << "!!!!\n";
34
35     return(0);
36 }
37
38 //Gets scores from judges. Returns Winner name and score
39 pair<string, double> getScores() {
40     int score1, score2, score3, score4, score5; //5 judge scores
41     string contestant;//name of contestant
42     double avgScore; //Avg score after droppign lowest and highest score
43     pair<string, double> winner; //placeholder for contestant and avgScore
44 }
```

```
45     do {
46         //Get judge scores
47         contestant = getInput("Enter name of contestant: ", "ERROR:Enter a valid
         string");
48         score1 = getInput("Enter score 1(1-10): ", "Enter a valid number between
         1 - 10", 1, 10);
49         score2 = getInput("Enter score 2(1-10): ", "Enter a valid number between
         1 - 10", 1, 10);
50         score3 = getInput("Enter score 3(1-10): ", "Enter a valid number between
         1 - 10", 1, 10);
51         score4 = getInput("Enter score 4(1-10): ", "Enter a valid number between
         1 - 10", 1, 10);
52         score5 = getInput("Enter score 5(1-10): ", "Enter a valid number between
         1 - 10", 1, 10);
53
54         avgScore = calcAvgScore(score1, score2, score3, score4, score5);//
         Calculate Avg Score
55
56         ///Debug Output
57         if (debug) cout << "AvgScore: " << avgScore << endl;
58
59         if (avgScore > winner.second) //Determine if current contestant is the
         winner
60             winner = make_pair(contestant, avgScore);
61     } while (getSentinel() == 'y');
62
63     return winner;
64 }
65
66 //Validates input based on a range. returns int
67 int getInput(string questionToAsk, string errorMsg, int lowRange, int highRange)
68 {
69     double usrInput;
70     cout << questionToAsk;
71     while (!(cin >> usrInput) || usrInput < lowRange || usrInput > highRange || !
72         (checkInt(usrInput))) { //Loop until integer in the specified range is
         entered
73         cout << errorMsg << endl;
74         cin.clear();
75         cin.ignore(numeric_limits<streamsize>::max(), '\n');
76     }
77     return int(usrInput);
78 }
79
80 //Validates input, returns string
81 string getInput(string questionToAsk, string errorMsg) {
82     string usrInput;
83     cout << questionToAsk;
84     while (!(cin >> usrInput)) { //Loop until integer in the specified range is
         entered
85         cout << errorMsg << endl;
86         cin.clear();
```

```
85     cin.ignore(numeric_limits<streamsize>::max(), '\n');
86 }
87 return usrInput;
88 }
89
90 //Averages 5 scores, dropping lowest and highest
91 double calcAvgScore(int score1, int score2, int score3, int score4, int score5) {
92     int lowScore = findLowest(score1, score2, score3, score4, score5);
93     int highScore = findHighest(score1, score2, score3, score4, score5);
94
95     //Debug Output
96     if (debug)cout << "LowScore: " << lowScore << "\n";
97     if (debug) cout << "HighScore: " << highScore << "\n";
98
99     return (score1 + score2 + score3 + score4 + score5 - lowScore - highScore) / 3.0; //average score
100 }
101
102 //Finds smallest int from 5 variables
103 int findLowest(int score1, int score2, int score3, int score4, int score5) {
104     int lowScore = score1;
105     if (score2 < lowScore)
106         lowScore = score2;
107     if (score3 < lowScore)
108         lowScore = score3;
109     if (score4 < lowScore)
110         lowScore = score4;
111     if (score5 < lowScore)
112         lowScore = score5;
113
114     return lowScore;
115 }
116
117 //Finds highest int from 5 variables
118 int findHighest(int score1, int score2, int score3, int score4, int score5) {
119     int highScore = score1;
120     if (score2 > highScore)
121         highScore = score2;
122     if (score3 > highScore)
123         highScore = score3;
124     if (score4 > highScore)
125         highScore = score4;
126     if (score5 > highScore)
127         highScore = score5;
128
129     return highScore;
130 }
131
132 //validates sentinel input, then returns char value
133 char getSentinel() {
134     char varToReturn;
135     bool isValidInput = false;
```

```
136
137     // loop until a valid y or n char is entered
138     do {
139         cout << "\nEnter 'y' to add more contestants. Enter 'n' if you are finished: " ;
140         if (!(cin >> varToReturn) || (tolower(varToReturn) != 'y' && tolower
141             (varToReturn) != 'n')) {
142             cout << "\tERROR: Enter 'y' to add more contestants. Enter 'n' if
143                 you are finished\n";
144             cin.clear();
145             cin.ignore(numeric_limits<streamsize>::max(), '\n');
146         }
147         else {
148             isValidInput = true;
149         }
150     } while (!isValidInput);
151     return tolower(varToReturn);
152 }
153
154
155 //checks if double is a whole number.
156 bool checkInt(double num) {
157     if (floor(num) == num)
158         return true;
159     else
160         return false;
161 }
162 }
```