

```

1  using namespace std;
2
3  class test_ui {
4  private:
5      ui mainUI;           // The testing UI object
6      bool testOutputs();
7      void breakline();
8  public:
9      bool executeTests();
10 };
11
12 void test_ui::breakline() {
13     // Visual function, outputs an 80 character breakline
14     // Requiriements: N/A
15
16     // Output a newline for ease of use
17     cout << endl;
18
19     // Create 80 dashes
20     for (int i = 0; i <= 80; i++) {
21         cout << "-";
22     }
23
24     // Another newline, for luck
25     cout << endl;
26 }
27 bool test_ui::testOutputs() {
28     bool tempReturn = true;    // Whether the run succeeded
29     string askStudentName_output; // The output of askStudentName()
30     int getAnswer_output;      // The output of getAnswer()
31     progress testProgress;     // Testing progress object
32
33     // Set the values of our testing progress object
34     testProgress.setStudentName("Charlie Test");
35     testProgress.answerTrue();
36     testProgress.answerTrue();
37     testProgress.answerTrue();
38     testProgress.answerFalse();
39     testProgress.answerFalse();
40
41     try {
42         // Attempt to generate the proper output
43
44         // Test function genHeader()
45         cout << "Header generation" << endl;
46         cout << "-----" << endl;
47         mainUI.genHeader();
48         breakline();
49
50         // Test function genMainMenu()
51         cout << "Main menu generation" << endl;
52         cout << "-----" << endl;
53         mainUI.genMainMenu();

```

```
54     breakline();
55
56     // Test function askStudentName()
57     cout << "Ask student name" << endl;
58     cout << "-----" << endl;
59     askStudentName_output = mainUI.askStudentName();
60     cout << "The student name entered was: " << askStudentName_output << endl;
61     breakline();
62
63     // Test function askLevel()
64     cout << "Asking the level of the questions" << endl;
65     cout << "-----" << endl;
66     mainUI.askLevel();
67     breakline();
68
69     // Test function askOperation()
70     cout << "Asking the operation" << endl;
71     cout << "-----" << endl;
72     mainUI.askOperation();
73     breakline();
74
75     // Test function askQuestion()
76     cout << "Asking a question the first time" << endl;
77     cout << "-----" << endl;
78     mainUI.askQuestion(10, 5, 2, false);
79     breakline();
80
81     // Test function AskQuestion() (again)
82     cout << "Asking a question the second time" << endl;
83     cout << "-----" << endl;
84     mainUI.askQuestion(10, 5, 2, true);
85     breakline();
86
87     // Test function getAnswer()
88     cout << "Getting an answer" << endl;
89     cout << "-----" << endl;
90     getAnswer_output = mainUI.getAnswer();
91     cout << "The answer entered was: " << getAnswer_output << endl;
92     breakline();
93
94     // Test function echoStudentScores()
95     cout << "Outputting student scores" << endl;
96     cout << "-----" << endl;
97     mainUI.echoStudentScores(testProgress);
98     breakline();
99
100    // Test function answerEchoResult() (true)
101    cout << "Student answered question right" << endl;
102    cout << "-----" << endl;
103    mainUI.answerEchoResult(true);
104    breakline();
105
106    // Test function answerEchoResult() (false)
```

```
107     cout << "Student answered question wrong" << endl;
108     cout << "-----" << endl;
109     mainUI.answerEchoResult(false);
110     breakline();
111
112     // Test function answerChoiceNotValid()
113     cout << "Student's choice is invalid" << endl;
114     cout << "-----" << endl;
115     mainUI.answerChoiceNotVaild();
116     breakline();
117 }
118 catch (...) {
119     // If it breaks, tests failed
120     tempReturn = false;
121 }
122
123 // Return whether we succeeded
124 return tempReturn;
125 }
126
127 bool test_ui::executeTests () {
128     // API to allow the running of tests
129     bool tempReturn = true; // Whether the tests succeeded
130     tempReturn = testOutputs(); // Execute the test
131     return tempReturn; // Return whether it worked
132 }
```