CS 202 Homework 2

Bryan Beus

February 15, 2020

Source Code Link: `https://github.com/siddhartha-crypto/cs202/tree/master/hw2`

1 Design

1.1 Main

I intend to use Anglo-Saxon poetry projects, counting the various elements that are relevant to the code project.

I use the Miscellaneous.cpp/hpp file that makes a few common functions easier.

The `std::list<Type>` method is sufficient and simple to use for all test purposes.

1.2 Spelunking

I intend to keep this simple. I am interested in doing this project, as the teacher recommends this project and I am curious to understand more about `weak_ptr` methods.

The rough layout provided in the homework assignment is enough to get started.

2 Post Mortem

2.1 Main

The only challenge I had in this project was in understanding how Catch2.hpp functions. I returned to the supplemental lab and completed it for better understanding.

I was confused with entering the content into the `TEST\_REQUIRE` section, not having a `main()`, but this was quickly resolved with help from Dr. Metzgar.

All in all, I spent probably five hours or so on this project, and I was not under a sense of urgency, so my mind was often wandering during these five hours.

2.2 Spelunking

The Spelunking project was a very enjoyable project. That said, it was definitely a bear. I spent over two full days.

I learned during the course of this project that I should make a habit of writing pseudo-code more often. I tend to write in code-language first, even though I am only thinking through the pseudo-logical aspect.

I would progress faster if I didn't worry about the complexities of the language at first.

Once I had the initial layout mapped, getting everything to compile was not difficult. The `weak\_ptr` aspect of the project did add some complexity here, but there was nothing that was not doable. I do thoroughly enjoy this aspect of programming.

Once the project was compiling, that was when the real challenge set in.

In many, many places I had made logical errors (that would have been prevented, had I started with pseudo-code first). There were so many logical errors, I spent hours tracking down bugs, and often reintroduced bugs in this process.

Once I was able to get the last few logical errors resolved, everything else was a cinch.

Dr. Metzgar has also pointed out that I would benefit from a faster workflow, with respect to the aspects of VIM that are not as elegant as many free GUI interfaces. I agree with this sentiment, and am contemplating how best to change this aspect of my workflow.

3 Commit History

3.1 Total

```
75d7055 (HEAD -> master, origin/master, origin/HEAD) finish hw2
d1a343d remove old program from hw2
8b29c9a comment main project in hw2
a89d405 add first two tests in main in hw2
c2a447c move all code into TEST CASE for Catch assignment in hw2
e1e61ce add long description reprint option
e21991b comment cave in hw2
f17ce18 comment main() in spelunking in hw2
485f43b large debug user prompt
f64a339 debug Cave.hpp and Cave.cpp compilation in spelunking in
↪  hw2
1af0751 update main() in spelunking hw2
2f46d87 Cave.hpp and Cave.cpp compile in spelunking in hw2
9999256 initial layout in main() complete, debugging compilation
↪  process in spelunking hw2
db040df initiate main in spelunking hw2
fdb5b9f initial layout for Cave.cpp
0bfd1fa initial layout for spelunking Cave.cpp in hw2
9ba144a layout rough draft for connectrooms() in hw2 spelunking
0f137f9 develop lab 4 and cave.hpp in hw2
4b3576a develop spelunking project
02093d8 update to make average keep float decimal
82868c3 lab05 fix seeg and tellg
8069dd1 lab04 finished
3452653 lab06
15b09d9 reset Cave.cpp file
3091e7d create initial layout for cave in hw2
5f6bfeb rm program name in hw2
104fd19 clean layout for Cave project in hw2
3bbabea initiate cave project in hw2
6e19efd finish initial draft of main in hw2
08afbbb add medieval files, update value class to manage
↪  reporting values in hw2 main
81a91dc fix uninitialized integers in lab04 and hw2 main
f7cf56a create initial layout for hw2 main project
```

3.2 Main

```
d1a343d remove old program from hw2
8b29c9a comment main project in hw2
a89d405 add first two tests in main in hw2
c2a447c move all code into TEST CASE for Catch assignment in hw2
104fd19 clean layout for Cave project in hw2
6e19efd finish initial draft of main in hw2
08afbbb add medieval files, update value class to manage reporting values in hw2 main
81a91dc fix uninitialized integers in lab04 and hw2 main
f7cf56a create initial layout for hw2 main project
e4a1a73 initiate hw2
891e45c resolve char to size_t for pos variable in C++ version of conversion for temp, hw0
b6a94bd first commit
```

3.3 Spelunking

```
75d7055 (HEAD -> master, origin/master, origin/HEAD) finish hw2
e1e61ce add long description reprint option
e21991b comment cave in hw2
f17ce18 comment main() in spelunking in hw2
485f43b large debug user prompt
f64a339 debug Cave.hpp and Cave.cpp compilation in spelunking in hw2
1af0751 update main() in spelunking hw2
2f46d87 Cave.hpp and Cave.cpp compile in spelunking in hw2
9999256 initial layout in main() complete, debugging compilation process in spelunking hw2
db040df initiate main in spelunking hw2
fdb5b9f initial layout for Cave.cpp
0bfd1fa initial layout for spelunking Cave.cpp in hw2
9ba144a layout rough draft for connectrooms() in hw2 spelunking
0f137f9 develop lab 4 and cave.hpp in hw2
8069dd1 lab04 finished
15b09d9 reset Cave.cpp file
3091e7d create initial layout for cave in hw2
5f6bfeb rm program name in hw2
104fd19 clean layout for Cave project in hw2
3bbabea initiate cave project in hw2
```

4 Answers to Questions

- Type: `int *ip = new int[10];` Release Method: `delete [] ip;`
- Type: `int *ip = new int;` Release Method: `delete ip;`
- Type: `int *ip = (int *)malloc(10*sizeof(int));` Release Method: `delete ip;`
- Returns the item of index `3` in the provided string
- Pass the shared object directly. Function: `void foo(const Widget& name)\{\;`, call using `foo(*name)`.
- A nullptr is a feature that allows a pointer to point to nothing. This is essential to avoid undefined behavior. Ideally, every pointer should be initialized to nullptr.
- Double delete results in undefined behavior.
- No, a pointer does not. Often, the item that it points to will have some value in its array that indicates to the pointer that it is pointing at the end.
- Code or Text storage; Static Storage; Stack Storage or Automatic Storage; Free Store or Heap
- Code/Text storage is that which is stored in the code or text; static is that which is pulled from a static source, such as from a root aspect of the system; the Stack is a part of the operating system that offers designated storage for a program to manage its functionality, and this is where the FIFO/LIFO actions are important; the Free Store is an additional aspect of the memory storage that is more free for all and less managed (i.e. not necessarily FIFO/LIFO)
- A leak is when memory is allocated and not properly deleted. When this happens repeatedly, the total memory available to a software application is gradually consumed, and can cause undefined behavior or even crash the software or system

- Gregory Yorb wrote Hunt the Wumpus. Wind indicates the presence of the bottomless pit. Wumpuses have suction cup holders. The Wumpus is smelly. The narrator indicates when there are bats.

5 Sample Output

5.1 Main

```
                                        Push three values to the
                                         ↪  front of the list
================================================================================
                          Filename  Paragraphs       Lines
                           ↪  Words  Characters
            solomon-and-saturn.txt         201         834
             ↪  4499       25606
================================================================================
                          Filename  Paragraphs       Lines
                           ↪  Words  Characters
             dream-of-the-rood.txt          19         185
              ↪  1433        7861
================================================================================
                          Filename  Paragraphs       Lines
                           ↪  Words  Characters
                 the-wanderer.txt          18         152
                  ↪  1011        5715

                                        Pop a value from the
                                         ↪  front of the
                                         ↪  list
================================================================================
                          Filename  Paragraphs       Lines
                           ↪  Words  Characters
             dream-of-the-rood.txt          19         185
              ↪  1433        7861
================================================================================
                          Filename  Paragraphs       Lines
                           ↪  Words  Characters
                 the-wanderer.txt          18         152
                  ↪  1011        5715

                                        Push three values to the
                                         ↪  back of the list
```

```
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
            dream-of-the-rood.txt          19         185
            ↪  1433        7861
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
                the-wanderer.txt          18         152
                ↪  1011        5715
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
           solomon-and-saturn.txt         201         834
           ↪  4499       25606
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
                 the-phoenix.txt          61         679
                 ↪  5379       29968
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
            wulf-and-eadwacer.txt           7          28
            ↪  147         797

                                      Pop a value from the back
                                      ↪  of the list
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
            dream-of-the-rood.txt          19         185
            ↪  1433        7861
================================================================================
                          Filename  Paragraphs       Lines
                          ↪  Words  Characters
                the-wanderer.txt          18         152
                ↪  1011        5715
```

```
================================================================================
                           Filename  Paragraphs       Lines
                           ↪  Words  Characters
            solomon-and-saturn.txt         201         834
            ↪  4499       25606
================================================================================
                           Filename  Paragraphs       Lines
                           ↪  Words  Characters
                  the-phoenix.txt          61         679
                  ↪  5379       29968

                                                    Search for
                                                    ↪  Solomon
                                                    ↪  and
                                                    ↪  Saturn
================================================================================
                           Filename  Paragraphs       Lines
                           ↪  Words  Characters
            solomon-and-saturn.txt         201         834
            ↪  4499       25606

                                    Insert sorted values behind
                                    ↪  Solomon and Saturn
================================================================================
                           Filename  Paragraphs       Lines
                           ↪  Words  Characters
       the-fate-of-the-apostles.txt          18         154
       ↪  997        5691
================================================================================
                           Filename  Paragraphs       Lines
                           ↪  Words  Characters
             dream-of-the-rood.txt          19         185
             ↪  1433        7861
================================================================================
                           Filename  Paragraphs       Lines
                           ↪  Words  Characters
                 the-wanderer.txt          18         152
                 ↪  1011        5715
```

```
==========================================================================================
                            Filename  Paragraphs       Lines
                            ↪  Words  Characters
              wulf-and-eadwacer.txt           7          28
              ↪  147         797
==========================================================================================
                            Filename  Paragraphs       Lines
                            ↪  Words  Characters
            solomon-and-saturn.txt         201         834
            ↪  4499       25606
==========================================================================================
                            Filename  Paragraphs       Lines
                            ↪  Words  Characters
                the-menologium.txt          66         319
                ↪  1774       10136
==========================================================================================
                            Filename  Paragraphs       Lines
                            ↪  Words  Characters
                   the-phoenix.txt          61         679
                   ↪  5379       29968
==========================================================================================
                            Filename  Paragraphs       Lines
                            ↪  Words  Characters
         the-order-of-the-world.txt          20         134
         ↪  869        4941

Total number of file changes: 10
==========================================================================================
All tests passed (9 assertions in 1 test case)
```

5.2 Spelunking

```
Current Room: 0
long description for room id 0
Adjacent Rooms:
Choice 0) short description for room id 1

Choice 1) short description for room id 2

```

```
Choice 2) short description for room id 3


Make a choice (0, 1, or 2) of the next room to visit.
Enter 3 to see the Long Description for this room
Enter 4 to perform a save demonstration and quit.
4

short description for room id 0
short description for room id 1
short description for room id 2
short description for room id 3
short description for room id 4
short description for room id 5
short description for room id 6
short description for room id 7
short description for room id 8
short description for room id 9
short description for room id 10
short description for room id 11
short description for room id 12
short description for room id 13
short description for room id 14
short description for room id 15
short description for room id 16
short description for room id 17
```

5.3 save_file.txt from Spelunking

```
long description
short description
0
1
2
3
long description
short description
1
0
2
3
long description
short description
2
1
3
4
long description
short description
```

```
3
2
4
5
long description
short description
4
3
5
6
long description
short description
5
4
6
7
long description
short description
6
5
7
8
long description
short description
7
6
8
9
long description
short description
8
7
9
10
long description
short description
9
8
10
11
long description
short description
10
9
11
12
long description
short description
11
10
12
13
long description
```

```
short description
12
11
13
14
long description
short description
13
12
14
15
long description
short description
14
13
15
16
long description
short description
15
14
16
17
long description
short description
16
14
15
17
long description
short description
17
14
15
16
```

---

6 My Programs

6.1 Main

```
/*
 * main.cpp
 * CS 202
 * February 14, 2020
 * Bryan Beus
 * Main file for main project in hw2
 */

#define CATCH_CONFIG_MAIN

#include <iostream>
#include <iomanip>
#include <string>
#include <vector>
#include <list>
#include <iterator>

#include "Miscellaneous.hpp"
#include "Value.hpp"
#include "Catch.hpp"

using std::cout;
using std::cin;
using std::endl;
using std::vector;
using std::string;
using std::list;
using std::right;

// Report the values for each list item in mylist
void reportValues(list<Value>& mylist) {

    list<Value>::iterator it = mylist.begin();

    for (unsigned int i = 0; i < mylist.size(); i++) {
        it->reportValue();
        it++;
    }

}

// Provides main() using Catch2.hpp
TEST_CASE( "Linked List/Stack tests", "[list]" ) {

    // Create vector to hold each filename
    vector<string> filenames;
    filenames.push_back("the-wanderer.txt");
```

```
48      filenames.push_back("dream-of-the-rood.txt");
49      filenames.push_back("solomon-and-saturn.txt");
50      filenames.push_back("the-phoenix.txt");
51      filenames.push_back("wulf-and-eadwacer.txt");
52      filenames.push_back("the-menologium.txt");
53      filenames.push_back("the-order-of-the-world.txt");
54      filenames.push_back("the-fate-of-the-apostles.txt");
55
56      list<Value> mylist;
57
58      // Demonstration of mylist.push_front() for three values
59      cout << endl;
60      cout << setw(35 + 12 * 4) << right << "Push three values to
        ↪  the front of the list" << endl;
61      for (unsigned int i = 0; i < 3; i++) {
62          Value newValue(filenames[i]);
63          mylist.push_front(newValue);
64      }
65      reportValues(mylist);
66      cout << endl;
67
68      // Demonstration for popping a value from mylist
69      cout << setw(35 + 12 * 4) << right <<  "Pop a value from the
        ↪  front of the list" << endl;
70      mylist.pop_front();
71      reportValues(mylist);
72      cout << endl;
73
74      // Demonstration of pushing to the back of the lsit
75      cout << setw(35 + 12 * 4) << right <<  "Push three values to
        ↪  the back of the list" << endl;
76      for (unsigned int i = 2; i < 5; i++) {
77          Value newValue(filenames[i]);
78          mylist.push_back(newValue);
79      }
80      reportValues(mylist);
81      cout << endl;
82
83      // Demonstration of popping from the back of the list
84      cout << setw(35 + 12 * 4) << right <<  "Pop a value from the
        ↪  back of the list" << endl;
85      mylist.pop_back();
86      reportValues(mylist);
87      cout << endl;
88
89      // Demonstration of sorting the list
90      list<Value>::iterator it = mylist.begin();
91      cout << setw(35 + 12 * 4) << right <<  "Search for Solomon and
        ↪  Saturn" << endl;
92      while (it->title != "solomon-and-saturn.txt") it++;
93      it->reportValue();
94
```

```
95      // Demonstration of inserting sorted values into the list
96      cout << endl;
97      cout << setw(35 + 12 * 4) << right <<  "Insert sorted values
        ↪  behind Solomon and Saturn" << endl;
98      for (int i = 0; mylist.size() < filenames.size(); i++) {
99          Value newValue(filenames[(int)(mylist.size())]);
100         mylist.insert(it, newValue);
101         it++;
102     }
103     reportValues(mylist);
104
105     // If each action performed properly, tests should be positive
106     list<Value>::iterator it_test = mylist.begin();
107     REQUIRE( it_test->title == "the-fate-of-the-apostles.txt" );
108     std::advance(it_test, 1);
109     REQUIRE( it_test->title == "dream-of-the-rood.txt" );
110
111     std::advance(it_test, 1);
112     REQUIRE( it_test->title == "the-wanderer.txt" );
113
114     std::advance(it_test, 1);
115     REQUIRE( it_test->title == "wulf-and-eadwacer.txt" );
116
117     std::advance(it_test, 1);
118     REQUIRE( it_test->title == "solomon-and-saturn.txt" );
119
120     std::advance(it_test, 1);
121     REQUIRE( it_test->title == "the-menologium.txt" );
122
123     std::advance(it_test, 1);
124     REQUIRE( it_test->title == "the-phoenix.txt" );
125
126     std::advance(it_test, 1);
127     REQUIRE( it_test->title == "the-order-of-the-world.txt" );
128
129     cout << endl;
130     cout << "Total number of file changes: " <<
        ↪  Value::total_change_count << endl;
131
132 }
```

6.2 Spelunking

```
1  /*
2   * main.cpp
3   * CS 202
4   * February 11, 2020
5   * Bryan Beus
6   * Main file for spelunking project hw2
7   */
```

```
8
9  #include <iostream>
10 #include <iomanip>
11 #include <string>
12 #include <vector>
13 #include <list>
14 #include <iterator>
15 #include <memory>
16 #include <fstream>
17 #include <sstream>
18
19 #include "Miscellaneous.hpp"
20 #include "Cave.hpp"
21
22 using std::cout;
23 using std::cin;
24 using std::endl;
25 using std::vector;
26 using std::string;
27 using std::list;
28 using std::right;
29 using std::ifstream;
30 using std::ofstream;
31 using std::istream;
32 using std::getline;
33 using std::istringstream;
34
35 int main(int argc, char* argv[])
36 {
37
38     // Declare new cave
39     Cave cave;
40
41     // Create a string that holds a default cave
42     string def_cave = cave.createDefaultCave();
43
44     // Read in the default cave
45     istringstream default_cave(def_cave);
46     cave.readRooms(default_cave);
47
48
49     // Initiate user input while loop
50     while (true) {
51         clearConsole();
52
53         // Discover current room
54         int currentRoom = cave.getCurrentRoom();
55         cout << "Current Room: " << currentRoom << endl;
56
57        // Print the long or short description of the current cave
58         if (cave.getVisited(currentRoom)) {
59             cave.printShortDescription(currentRoom);
60         } else {
```

```
61              cave.printLongDesc(currentRoom);
62          }
63
64          // Discover adjacent rooms
65          vector<int> adjacent_rooms =
            ↪  cave.getAdjacentRooms(currentRoom);
66          cout << "Adjacent Rooms: " << endl;
67
68          // Present user with description of adjacent rooms and
            ↪  choices
69          for (int i = 0; i < 3; i++) {
70              cout << "Choice " << i << ") ";
71              cave.printShortDescription(adjacent_rooms.at(i));
72              cout << endl;
73          }
74
75          // Caputre user input
76          int userInput;
77          capture_user_input(userInput);
78
79          // Quit if user has indicated
80          if (userInput == 3) {
81              cave.printLongDesc(currentRoom);
82          } else if (userInput == 4) {
83              break;
84          } else {
85          // Proceed to adjacent room
86          cave.gotoAdjacentRoom(adjacent_rooms.at(userInput));
87          }
88      }
89
90      // Save default cave to file (for proof-of-concept)
91      ofstream fout("save_file.txt");
92
93      if (!fout) {
94          cout << "File save failed" << endl;
95          exit(0);
96      }
97
98      cave.saveRooms(fout);
99      fout.close();
100
101     // Read in the saved file, to demonstrate proof of concept
102     Cave newCave;
103
104     ifstream fin("save_file.txt");
105
106     if (!fin) {
107         cout << "Opening file failed" << endl;
108         exit(0);
109     }
110
111     newCave.readRooms(fin);
```

```
112
113     // Pring short descriptions to prove functionality
114     for (int i = 0; i < 18; i++) {
115         cave.printShortDescription(i);
116     }
117
118     return 0;
119 }
```