Week 3 Challenge

LATEST SUBMISSION GRADE

100%

1. A class called Pair has already been declared, but the constructors have not been implemented yet. Pair has two public member variables:

5 / 5 points

int *pa,*pb;

These two "pointers to int" are intended to point to heap memory locations that store integers. The remainder of the Pair class expects the following functionality.

- A single constructor Pair(int a, int b): This should set up pa and pb to point to newly allocated memory locations on the heap. The integers at those memory locations should be assigned values according to the constructor's integer arguments a and b.
- A copy constructor Pair(const Pair& other): This takes as its argument a read-only reference to another Pair. It should set up the newly constructed Pair as a "deep copy," that is, it should create a new Pair that is equivalent to the other Pair based on dereferenced values but does not reuse any of the same memory locations. In other words, the copy constructor should set up its own instance's member variables pa and pb to point to newly allocated memory locations for integers on the heap; those memory locations must be new, not the same locations pointed to by the other Pair, but the integers at these new locations should be assigned values according to the integers that the other Pair is pointing to.
- A destructor ~Pair() that de-allocates all of the the heap memory that had previously been allocated for this Pair's members.

The types of these member functions have already been declared in the declaration of Pair. Now you need to provide the implementation of each of these three member functions.

(Note: The function declarations shown in the code comment below do not include parameter names for the arguments. They show only the types of the arguments. This is allowed for a declaration, but when you define the implementation of those functions, you should give names to the parameters so that you can refer to them.)

```
 * };
 *
 * Implement its member functions below.
 */
 Pair::Pair(int a, int b) {
  pa = new int;
  pb = new int;
  *pa = a;
  *pb = b;
 }

 Pair::Pair(const Pair& other) {
  pa = new int;
  pb = new int;
  *pa = *(other.pa);
  *pb = *(other.pb);
 }

Pair::~Pair() {
  delete pa;
  delete pb;
  delete pb;
}


 /* Here is a main() function you can use
  * to check your implementation of the
  * class Pair member functions.
  */

int main() {
  Pair p(15,16);
  //Pair q(p);
  //Pair *hp = new Pair(23,42);
  //delete hp;

  std::cout << "If this message is printed,"
    << " at least the program hasn't crashed yet!\n"
    << "But you may want to print other diagnostic messages too." << std::endl;
  return 0;
}
```

Run

Reset

✓ **Correct**

Passes all tests!