HW2 OOP-dry part

Irad Nuriel, 324220458

Question 1:

the output is 2.

explanation:

After the first row in main, b1 is instance of B that by default default constructor of B, contains instance of A that by default default constructor of A, contains int pointer which by default is null.

After the second row in main, the int pointer b1._a.x, pointing on 1.

After the third row of main, b2 is instance of B, that by default copy constructor, contains instance of A that because the default copy constructor of B is bit-wise copy, contains a pointer to the exact same place in the memory like the pointer b1._a.x(which contains 1).

After the forth row of main, the value in *b2._a.x is 2, and because b2._a.x and b2._a.x are pointing to the same place in memory, *b1._a.x also contains 2.

The fifth row of main, the value of *b1._a.x , which, as described before, is 2.

Question 2:

The bug in this solution is memory leak. that happen because class Complex should have the big 3(d'tor, copy c'tor and assignment operator) implemented(because there are dynamic memory allocation) but didn't done so. the fix will be easy by implementing those three.

row	c1 value	c2 value	c3 value	explanation
1	Complex(*(_real(id=4))=2, *(_imag(id=3))=4)	Complex(*(_real(id=2))=5, *(_imag(id=6))=3)	-	two regular constructors, the default value of imag in the constructor of c2 is 3.
2	Complex(*(_real(id=5)=6, *(_imag(id=6))=3)	Complex(*(_real(id=5))=6, *(_imag(id=6))=3)	-	c2.setReal(6) will set the value of *(c2._real) is 6, and the default assignment operator will bit-wise copy the pointers in c2 to the pointers in c1
3	Complex(*(_real(id=5)=6, *(_imag(id=6))=3)	Complex(*(_real(id=5))=6, *(_imag(id=6))=3)	Complex(*(_real(id=7))=12, *(_imag(id=8))=3)	the + operator will return a pointer to a new Complex that his _real will be 12, and his _imag will be 3(as default). then, the default copy constructor will be called, and, will bit-wise copy the pointers of the new Complex to c3(his fields will be pointers from the pointer returned from the + operator
4	Complex(*(_real(id=5)=6, *(_imag(id=6))=3)	Complex(*(_real(id=5)=6, *(_imag(id=6))=3)	Complex(*(_real(id=7))=12, *(_imag(id=8))=3)	no change in the values of c1,c2,c3.

The output is:

real is: 12, and imaginary is: 3

Question 3:

The output will be 0.

Explanation:

in the first two rows of main, container and container2 will be constructed by the default constructor of Container<Complex,7> and Container<Complex,5> respectively.

In the third row of main, the 0^{th} item in container will be set to a new Complex that equal (real=1 and imag=2). after that, the constructor Container<Complex,5>(Container<Complex,7>& other) will be called which will make container2 a new Container(size 5) with _arr = pointer to array of 5 default complexes(real=0 an imag = 0).

from that, the output is totally trivial because getItemAtIndex(0) will return Complex(real = 0, imag = 0) and (returned Complex).getReal() will return 0.