Answer the questions in Exercise A in the following table and post it into the D2L

Program output and its order	Your explanation (why and where is the cause for this output)
constructor with int argument is called.	It is called at line 12 in exAmain. The statement, Mystring c=3 is interpreted by the compiler as a call to the constructor Mystring::Mystring(int n).
default constructor is called. **default constructor is called.**	It is called at line 18 in exAmain. The statement, Mystring x[2] is interpreted by the compiler as call to the default constructor Mystring::Mystring() , and since it creates two objects the output is shown twice.
constructor with char* argument is called.	It is called at line 22 in exAmain. The statement, Mystring* z = new Mystring("4"); is interpreted by the compiler as a call to the constructor with char* argument Mystring::Mystring(const char *s);
copy constructor is called. **copy constructor is called.**	It is called in line 24 in exAmain. The statement x[0].append(*z).append(x[1]) has to first create a copy of the object z is pointing to as the argument to do this it calls the copy constructor. The append gets called again and has to create another copy of x[1] hence why there are two copy constructor messages called
destructor is called. **destructor is called.**	It is called after line 24 in exAmain. This is because the *z and the x[1] copies are now out of scope, hence why the destructor was called twice.
copy constructor is called.	This is called on line 26 in exAmain. The statement Mystring mars = x[0]; has to create a copy of x[0] first, to ensure that nothing in x[0] gets changed. Then the values of x[0] are copied into mars.
assignment operator called.	This is called on line 28 in exAmain. The statement x[1] = x[0] calls the assignment operator Mystring& Mystring::operator = (const Mystring& S).
constructor with char* argument is called. **constructor with char* argument is called.**	These are called on line 30 and 32 when Jupiter is created and the pointer ar[0] points to the newly created Mystring object.
destructor is called. **destructor is called.** **destructor is called.** **destructor is called.** **destructor is called.**	This is called on line 34 and in line 37 in exAmain. x[0], x[1], mars Jupiter are deleted and destructor is called because it is out of scope, and ar[0] gets deleted because of delete being called on line 37 in exAmain.
constructor with char* argument is called.	It is called at line 39 in exAmain. The statement, Mystring

	d = “Green”; is interpreted by the compiler as a call to the constructor with char* argument Mystring::Mystring(const char *s);
Program terminated successfully.	This is called on line 41 because of the cout statement and the rest of the code is done
destructor is called. **destructor is called**	This is called on line 43 where the last bracket is, c and d are deleted because main() has ended, thus they are no longer in scope.