Exercise A

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.	When line 18 in exAmain, the default constructor is called twice and these outputs where created. Since x is an array of Mystring objects, the default constructor is called twice: once for x[0] and once for x[1].
constructor with char* argument is called.	At line 22 in exAmain, Mystring* z = new Mystring("4"); This line dynamically allocates a new Mystring object on the heap and initializes it with the string "4". The constructor Mystring::Mystring(const char* s) is called with the C-string "4" as the argument.
copy constructor is called. copy constructor is called.	At line 24 in exAmain, x[0].append(*z).append(x[1]); copy constructor is called each time the append function is called. When the Mystring objects are passed by value to append function, a copy of the object is created and passed to the function. The copy constructor is automatically called to create this copy of the object.
destructor is called. destructor is called.	Destructor is called to destroy the copy of the Mystring objects created in the append function call at line 24 in exAmain.
copy constructor is called.	At line 26 in exAmain, Mystring mars = x[0]; The Mystring object x[0] is passed to the Mystring copy constructor to initialize mars.
assignment operator called.	Line 28 in exAmain, x[1] = x[0]; uses the assignment operator to copy the string from x[0] to x[1]
constructor with char* argument is called. constructor with char* argument is called.	At lines 30 & 32 in exAmain, constructor Mystring::Mystring(const char *s) gets called to create Mystring objects with c-strings "White" and "Yellow" as arguments.

destructor is called. destructor is called. destructor is called. destructor is called. destructor is called.	When the block ends on line 34 in exAmain, the destructor will be invoked for objects(x[0], x[1], mars, and jupiter) that go out of scope. The dynamically allocated Mystring object pointed to by z and the dynamically allocated Mystring object assigned to ar[0] are not automatically destroyed when the block ends because they are managed with raw pointers Also, when line 37, delete ar [0]; is executed, it calls the destructor of the Mystring object pointed to by ar[0].
constructor with char* argument is called.	At line 39 in exAmain, Mystring d = “Green" The constructor Mystring(const char *s) is called with "Green" as the argument.
Program terminated successfully.	Line 41 of exAmain, outputs a message to the standard output stream.
destructor is called. destructor is called	Line 43 of exAmain, the destructor will be invoked for objects(c & d) that go out of scope.