MINIA UNIVERSITY
FACULTY SCIENCE
Department of Computer Science
Data Structures Using Python

Exercises #6
ArrayStack & ArrayQueue Classes

6.1 Write a segment of code to perform each of the following operations. Assume myStack is an object of the class ArrayStack. You may call any of the methods of ArrayStack class. You may declare additional stack objects.

a. Set *secondElement* to the second element in myStack, leaving myStack without its original top two elements.

b. Set *bottom* equal to the bottom element in myStack, leaving myStack empty.

c. Set *bottom* equal to the bottom element in myStack, leaving myStack unchanged.

d. Make a copy of myStack, leaving myStack unchanged.

6.2 Assume you have a list that contains integers stored in random order. Using the class ArrayStack, write a program that takes out all the integers from the list, one by one, discards the odd integers, and keeps the even integers in order, then returns them to the list in exactly the same order as before—without the odd integers, of course.

6.3 Write a program that prompts the user to enter a positive integer and displays all its prime factors in decreasing order. For example, if the integer is 120, the prime factors are displayed as 5, 3, 2, 2, 2. Use the class ArrayStack to store the factors (e.g., 2, 2, 2, 3, 5) and retrieve and display them in reverse order.

6.4 Write a segment of code to perform each of the following operations. Assume myQueue is an object of the class ArrayQueue. You may call any of the methods of ArrayQueue class. You may declare additional queue objects.

a. Set *secondElement* to the second element in myQueue, leaving myQueue without its original front two elements.

b. Set *last* equal to the rear element in myQueue, leaving myQueue empty.

c. Set *last* equal to the rear element in myQueue, leaving myQueue unchanged.

d. Make a copy of myQueue, leaving myQueue unchanged.

6.5 Using the class ArrayQueue, with its methods, write a main program that reads a string from the keyboard, which consists of two parts separated by a colon ':', then displays one of the following messages:

The left part (before the colon) is longer than the right part.
The right part (after the colon) is longer than the left part.
The left and right parts have the same length but are different.
The left and right parts are exactly the same.

(Hint: use a queue to keep the left part of the string while processing the right part.)