

# school of **computing, informatics, & decision systems engineering**

## CSE 110 – Assignment #7

Maximum points: 20 pts

### Topics

- Array (Chapter 6)
  - Array object as an instance variable
  - Partially filled array
  - Basic array operations (add, search, and delete)

Your programming assignments require **individual work** and effort to be of any benefit. Every student must work independently on his or her assignments. This means that every student must ensure that neither a soft copy nor a hard copy of their work gets into the hands of another student. Sharing your assignments with others in any way is **NOT** permitted. Violations of the University Academic Integrity policy will not be ignored. The university academic integrity policy is found at <http://www.asu.edu/studentlife/judicial/integrity.html>

### Use the following Guidelines:

- Give identifiers semantic meaning and make them easy to read (examples numStudents, grossPay, etc).
- Keep identifiers to a reasonably short length.
- User upper case for constants. Use title case (first letter is upper case) for classes. Use lower case with uppercase word separators for all other identifiers (variables, methods, objects).
- Use tabs or spaces to indent code within blocks (code surrounded by braces). This includes classes, methods, and code associated with ifs, switches and loops. Be consistent with the number of spaces or tabs that you use to indent.
- Use white space to make your program more readable.

### Important Note:

All submitted assignments must begin with the descriptive comment block. To avoid losing trivial points, make sure this comment header is included in every assignment you submit, and that it is updated accordingly from assignment to assignment. **(If not, -1 Pt)**

```
//*****  
// Name: your name  
// Title: title of the source file  
// Author: (if not you, put the name of author here)  
// Description: Write the description in your words.  
// Time spent: how long it took you to complete the assignment  
// Date: the date you programmed  
//*****
```

### Part# 1: Writing Exercise: (5 pts)

Consider the following array:

```
int[] a = { 2, 4, 6, 8, 10, 12};
```

Write the contents of the array **a** after the following loops, and explain the reason in your words. Use the original data above for each question. (1 Point each)

- a) for (int i = 1; i < 6; i++) { a[i] = a[i - 1]; }
- b) for (int i = 5; i > 0; i--) { a[i] = a[i - 1]; }
- c) for (int i = 0; i < 5; i++) { a[i] = a[i + 1]; }
- d) for (int i = 4; i >= 0; i-=2) { a[i] = a[i + 1]; }
- e) for (int i = 1; i < 6; i++) { a[i] = a[5 - i]; }

**Note:** The answers to the 5 questions (a through e) above should be typed in the block of comments in the Numbers.java file such as;

```
/*  
    a) {2, 4, 6, 8, 10, 12}: The array does not change anything  
        because...  
    b) ...  
    ...  
*/
```

## Part#2: Programming (15 pts)

Your assignment is to write and submit two java files. One is a class definition named **Quiz** (Quiz.java), and another is the test driver program, **Assignment7.java**.

1) The Quiz is a class to keep the scores of quiz for students, and it has the three instance variables: scores, count, and name.

Quiz
- scores : int[]
- count : int
- String : name
+ Quiz(int, String)
+ add(int) : void
+ delete (int) : void
+ print() : void
- search(int) : int

The count is the number of quizzes, 2) scores is an array object storing all scores as integers, and 3) name is a string object for student name.

The class Quiz must include the constructors and methods in the table: (If your class does not contain any of the following methods, points will be deducted). The public and private access modifiers are represented as "+" and "-" respectively.

1. Quiz(int, String): (2 pts)

The constructor is to

- Create an instance of an int-type array with the length of input parameter. Each element in the array is initialized as -1.
- The count is set as 0. It is not the length of array but the number of valid scores. In other words, the scores is a partially filled array, and the count is used as an END position of valid scores.
- Set the name with the second input parameter.

2. add(int):void (2 pts)

If there is room (left in) the array, add the parameter value to the END(count) position of the array and adjust the count to reflect how many values are in the array. If the value was added, DO NOT print any output. If the array is full, print the error message as shown below. This is the output if the value 3 was to be added and there is no room.

Array is full. The value 3 cannot be added.

3. `print(): void` (2 pts)

Print the name of student and the valid quiz scores in the array as shown below. Note that it prints at most 5 values per line and that the values are separated by three spaces. If the values {3, 6, 7, 4, 1, 4, -1, -1, -1} are in the array, then the output is

```
James Bond
3   6   7   4   1
4
```

4. `search(int):int` (2 pts)

This method should perform a linear or sequential search. The method is to look at the elements in the array until the value of the parameter is found or all values have been examined. If a value occurs in the array more than once, the method should return the index, first position, where the value was found. If the value is not in the array, then the method should return a -1. There is NO output from this method. (no printing)

5. `delete(int):void` (2 pts)

Search for the value in the array using the search method. If the value is currently in the array, remove the value from the array by moving the other elements “up” and adjust the count to reflect how many values are in the array. If the value was successfully removed, do not print any output. Here is the array after removing 6 {3, 7, 4, 1, 4, -1, -1, -1}. Only if the value is not found, then print the error message as shown below. It is an example case when the value 2 was to be deleted from the array {3, 6, 7, 4, 1, 4, -1, -1, -1}.

The value 2 was not found and cannot be deleted.

2) Write the test driver called [Assignment7.java](#). (5 pts) Assignment7.java contains the main method to create a Quiz instance object and test the methods in the class. The program will ask a user to enter one of the commands. Based on the user's choice, the program needs to perform corresponding operation. This will be done by calling (invoking) one of the methods you defined in the Quiz class. It will terminate when the user enters 'q'. The tester program provided in the last assignments may help. In addition to the main method, the Assignment7 class has a static method, `printMenu()`, to print out the following commands.

```
Command Options
-----
n: Create a new data
a: Add a score
d: Delete a score
p: Print the information
?: Display the menu again
q: Quit this program
```

Here is the description for each option:

n: ask and read the size of array and the name of student

a: ask and add a score to the array

d: ask and delete a score in the array

p: print the information of array.

?: displays the menu

q: quits the program

.

## Output Example

### Program Output (Input in red)

```
***** Quiz Tester Program *****

Command Options
-----
n: Create a new data
a: Add a score
d: Delete a score
p: Print the information
?: Display the menu again
q: Quit this program

Please enter a command or type ?: n
    n [Create a new Quiz]
        [Input the size of quizzes]: 6
        [Input the name of student]: James Bond
Please enter a command or type ?: a
    a [Add a score]: 5
Please enter a command or type ?: a
    a [Add a score]: 4
Please enter a command or type ?: a
    a [Add a score]: 3
Please enter a command or type ?: a
    a [Add a score]: 2
Please enter a command or type ?: a
    a [Add a score]: 1
Please enter a command or type ?: a
    a [Add a score]: 0
Please enter a command or type ?: a
    a [Add a score]: 3
    Array is full. The value 3 cannot be added.
Please enter a command or type ?: P
    p [Print the information]:

    James Bond
    5   4   3   2   1
```

0

```
Please enter a command or type ?: d
      d [Delete a score]: 3
Please enter a command or type ?: d
      d [Delete a score]: -1
      The value -1 was not found and cannot be deleted.
Please enter a command or type ?: d
      d [Delete a score]: 5
Please enter a command or type ?: p
      p [Print the information]:

      James Bond
      4      2      1      0
```

```
Please enter a command or type ?: ?
Command Options
-----
n: Create a new data
a: Add a score
d: Delete a score
p: Print the information
?: Display the menu again
q: Quit this program
```

```
Please enter a command or type ?: q
***** End of Program *****
```

Use only the Java statements that have been covered in class to date. **DO NOT** use any other items (array, array-list, split(), etc.) out of the Chapter1-5 and 8. If in doubt, ask. If you use them, then you lose the points of task. Don't copy any code developed by others. Don't give your code to others. Don't use any algorithm, which you cannot understand. Your assignment file is checked by the MOSS (by Stanford Univ.), which is a program to detect cheatings.

/\*\*\*\*\*

### **Submit your homework by following the instructions below:**

\*\*\*\*\*/

- Go to the course web site ([my.asu.edu](http://my.asu.edu)), and then click on the on-line Submission tab. Log in the site using the account, which was registered at the first Lab session. Make sure you use the correct email address for registration. This will allow you to submit assignments. Please use your ASU e-mail address.
- Submit your **Assignment7.java** and **Quiz.java** files on-line. Make sure to choose **HW7** from drop-down box.
- The **Quiz.java** should have the following, in order:
  - In comments, the assignment Header described in "Important Note".
  - In comments, the answers to questions presented in Part#1.
  - The working Java code requested in Part #2.
  - The **Assignment7.java** and **Quiz.java** files must compile and run as you submit it. You can confirm this by viewing your submission results.

**Important Note:** You may resubmit as many times as you like until the deadline, but we will only mark your last submission. **NO LATE ASSIGNMENTS WILL BE ACCEPTED.**