

# Discussion 2 handout

## Requirements reminder

- Form a group of 2-4 classmates seated near enough to allow discussion. Part of the purpose of discussion sections is for you to learn to work *collaboratively* on technical problems (software is written by teams).
- Record the group's responses to each activity on a sheet of paper (the boxes on this handout highlight expected responses). Write the NetIDs of all group members at the top.
- Register your group in CMSX, then upload a photo of the work you completed in section by *Friday evening*.
- It is okay if you do not finish everything during section. Please work at the pace facilitated by your TA (discussion is not a race).

## Group members (names & NetIDs)

- 1.
- 2.
- 3.
- 4.

## Objectives

- Draw object diagrams to illustrate runtime state and reason about reference semantics
- Write Java code to create and process arrays
- Process program arguments in `main()`

## Problem 1: Object diagrams

**Draw** labeled boxes for variables "x", "y", "a", and "b". Then **execute** the following assignment statements by hand, adding and erasing values and arrows in your diagram accordingly.

```
x = 3;  
y = 2 * x;  
x = 7;  
z = y;  
  
a = "world";  
b = a;  
a = "hello";
```

x: int	<input type="text"/>	a: String	<input type="text"/>
y: int	<input type="text"/>	b: String	<input type="text"/>
z: int	<input type="text"/>		

## Problem 2: Multidimensional arrays

**Draw** an object diagram to illustrate execution of the following statements:

```
int[][] a = new int[3][];  
a[0] = new int[]{3, 1, 4};  
a[1] = new int[]{1, 5};  
a[2] = a[0];  
a[2][1] = 2;
```

Recall the semantics of the assignment statement from last week's discussion. Based on that, what should happen when the following statement is executed?

```
a[1][2] = 9;
```

## Problem 3: Program arguments

**Implement** an “echo” program that prints each argument passed to it, separated by spaces. Starter code is provided on the [“dis02” project archive](#) on the course website.

Example:

```
$ java Echo My name is Hal.  
My name is Hal.
```

```
public class Echo {  
    public static void main(String[] args) {  
  
  
  
  
  
  
  
  
  
    }  
}
```

## Submission

1. Open the assignment page for “Discussion activity 2” in CMSX
2. [Recorder] Find the “Group Management” section and invite each group member
3. [Others] Refresh the page and accept your invitation
4. [Recorder] Take a picture of your work and save as either a JPEG or a PDF file named “discussion\_responses”. *After all invitations have been accepted*, upload your picture along with your code as your group’s submission.
  - Recommended scanning apps: Microsoft Office Lens, Adobe Scan, Genius Scan, Evernote Scannable

Ensure that your group is formed and your work submitted before the Friday evening deadline.

## Tips and reminders

- Discussion is not a race. Focus on the current activity being facilitated by your TA and engage your whole group to propose and explain ideas.
  - Elect a recorder to maintain the “master” copy of your work (others are still encouraged to jot down ideas on scratch paper). Rotate this position each week.
  - It is a violation of academic integrity to credit someone for work if they were not present when the work was done, and the whole group is accountable. Your CMS group must only include classmates who attended section with you on that day. Remember that our participation policies accommodate occasional absences without penalty.
  - It is your individual responsibility to ensure that you are grouped on CMS and that your group’s work is submitted before the deadline. Log into CMS the day after section to ensure that everything is in order, and contact your groupmates if not. It would be prudent for multiple members to photograph the group’s work.
  - Only one group member (typically the recorder) needs to upload a submission. But their submission must not be uploaded until after all group members have confirmed their membership in CMS (contact your TA if you have trouble grouping in CMS).
-