Mid-term Exam

- Date: 15th Nov, Wed
- Time: 8:30 ~ 11:00
- Venue: AB1087 (same as lectures)
- 20% of overall points

- Format:
  - Hand-writing in printed exam paper
  - Close book:
    - no computer/mobile allowed for on-site students
    - Specific requirement for on-line (remote) students, ensuring fairness across board

- Type of questions:
  - #1: multiple choice (with one or more than one correct answers/options)

Which of the following statements would cause a syntax (compilation) error? Assume all classes have a default constructor.

A. `Object obj = new Railroad();`
B. `Street s = new BoardSpace();`
C. `BoardSpace b = new Street();`
D. `Railroad r = new Street();`

- Type of questions:
  - #2: short conceptual questions
    - What is encapsulation in java?
    - What are the advantages of inheritance?

- Type of questions:
  - #3: identification of errors/exceptions in code
    - int a=50/0; //ArithmeticException
    - String s=null; System.out.println(s.length()); // NullPointerException

- Type of questions:
  - #4: read codes and figure out the output of a given java program
    - Assume the codes are compiled successfully.

```
public static void main(String args[])
{
  try {
        int data=25/0;
        System.out.println(data);
  }
  catch(ArithmeticException e)
  {
        System.out.println("exception is handled");
  }
  finally
  {
        System.out.println("in the finally block …");
  }

  System.out.println("rest of the code...");
}
```

- Type of questions:
  - #5: read the partially-given program/method, and then complete it according to requirements.
    - Like Q3 in Homework #3 – CreditCard.java
      - Shorter program

```
public class CreditCard
{

  public CreditCard(String cust, String bk, String acnt, int lim) {

    // call the previous constructor, but with an initial a balance of zero as default
    ... ...
  }


  // develop get/access methods for each attribute defined above (total 5)
  public ... get...() { ... ... }
```

- Type of questions:
  - #6: Write full programs/methods according to requirement
    - Clear goal/requirement
    - Likely shorter than your homework questions
      - *“Write a short Java method that takes an integer N and returns the sum of the squares of all positive integers less than or equal to N.”*

Any questions?

Topics for Midterm

- Basic java programming
- Object-oriented programming
- Exception handling
- Recursion

Topics for Midterm

- Primitive types
  - Boolean
    - operation with &&, ||
    - = vs ==
  - ASCII (alphabets representation)
- Array, two-dimensional array
  - Index, out of bound, initialization
- String, some typical methods
  - charAt(int), substring(int), etc.
- Method parameter: pass by value / by reference
  - String, array, class
- OOP:
  - Polymorphism (super/sub class, type declaration vs. binding)
  - Method signature: overloading / overriding
- Abstract class and interface:
  - Instantiation, extension/inheritance
  - Compare two objects
    - ==, equals
    - comparable interface for your own class

Topics for Midterm (continued)

- Exception handling
  - Error vs exception
  - Try, catch, finally
  - Customized exception, throws (see slides)

- Recursion
  - Method call and stack
  - vs. iteration (see slides on Fibonacci sequence)
  - Exercise:
    - Decimal to binary

Tips

- Go through slides, even those we didn’t cover in lectures
  - Not only for the revision and preparation of mid-term, but a good practice and further understanding of new concepts in java
- Write as clearly as you can
  - Failing to do so may render you unnecessary reduction of points
- Do not leave questions unanswered at all, especially those coding questions
  - Even just idea or flow of solving them.