COSC 111:
Computer Programming I

Dr. Bowen Hui

University of British Columbia
Okanagan

A1 feedback

- Solution online
- Statistics:
  - Average: 17/20
  - Max: 20/20
- Common mistakes:
  - When to initialize attributes using constructor's input parameters and when to give default values
  - Variables vs. values
  - Understanding values passed into input parameters
  - Use methods that are provided for you (e.g. setOrder())
  - Matching parameters between method calls and method definitions

Midterm – Things to bring

- Required:
  - Student ID card
- Allowed:
  - 2 pencils
  - An eraser
  - 2 pens
  - 1 sheet of 8.5” x 11” paper
- Not allowed:
  - No calculators of any kind
  - No cell phone
  - No food or drinks

Midterm instructions

- We will place the exam on a desk
  - Do not open the booklet
- Exam starts only when we indicate so
  - If anyone starts earlier, the case will be treated as cheating
- Exam ends when we indicate “time’s up”
  - If anyone stops later, the case will be treated as cheating
  - If you finish early, submit your exam before you pack up

Midterm instructions (cont.)

- No looking around, whispering, using any notes/books aside from the cheatsheet, copying from others, listening to audio, using the cell phone for any purpose
  - If any such behaviour is found, the case will be treated as cheating

- If you have a question, raise your hand during exam

Midterm Format

- Section 1: multiple choice [10 points]

- Section 2: short answers [15 points]
  - Given code, how many …?
  - Given code, identify the line of code that …
  - Given code, what is the output?

- Section 3: write Java code [22 points]
  - Given partial code, complete a method
  - Given partial code, write a method that does something specific

Questions?

Extra Practice

- Pseudo-random number generator implements a Math formula that generates a sequence of seemingly random numbers
- Example:

  $x_i = (p1 * x_0 + p2) \% N$

  where

  p1, p2 are constants

  N specifies the range of numbers to be returned in [0,N-1]

  x_0 is an initial number called "the seed"

  i = 1, 2, 3, …
- What kind of numbers are generated?

Extra Practice (cont.)

- Example:

```
int x0 = 1230128;
int p1, p2, N, xi;
p1 = 234;
p2 = 83;
N = 100;
xi = (p1 * x0 + p2) % N;
```

Extra Practice (cont.)

- Example:

```
int x0 = 1230128;
int p1, p2, N, xi;
p1 = 234;
p2 = 83;
N = 100;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    81
```

Extra Practice (cont.)

- Example:

```
int x0 = 1230128;
int p1, p2, N, xi;
p1 = 234;
p2 = 83;
N = 100;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    81
x0 = xi;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    37
```

Extra Practice (cont.)

- Example:

```
int x0 = 1230128;
int p1, p2, N, xi;
p1 = 234;
p2 = 83;
N = 100;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    81
x0 = xi;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    37
x0 = xi;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    41
```

Extra Practice (cont.)

- Example:

```
int x0 = 1230128;
int p1, p2, N, xi;
p1 = 234;
p2 = 83;
N = 100;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    81
x0 = xi;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    37
x0 = xi;
xi = (p1 * x0 + p2) % N;
System.out.println( xi );                    41
x0 = xi;
…
```

Extra Practice (cont.)

- How to write a Java program for the following number generator?

  $x_i = (p1 * x_0 + p2) \% N$

  where

  p1, p2 are constants

  N specifies the range of numbers to be returned in [0,N-1]

  x_0 is an initial number called "the seed"

  i = 1, 2, 3, …

- Hint:
  - Set up the variables inside the constructor
  - Do the first calculation inside the constructor
  - Define computeAgain() so the test class can repeatedly call it to get additional values of x_i