CSE 21
Intro to Computing II

Lecture 1 – General Course Information and Review of CSE20 (1)

CSE 21: Spring 2018

- Instructor
  - Santosh Chandrasekhar
  - schandrasekhar@ucmerced.edu
  - Office Room: AOA 143
  - Office Hours:
    - **T 1:00-5:00pm, R 1:00-3:00pm**
    - By appointment
- TA
  - See syllabus for names, office hours and location
- Email Policies
  - All email inquiries received before 5pm during school days will be replied within 48 hours
  - Please follow the guidelines available in CatCourses for proper email communications

Lecture Guidelines

- Your success is my success
  - This lecture is only successful when you understand the material being presented
- Please ask questions
  - Raise your hand, or just speak up
  - Don’t be shy, you are not being graded here
- Please speak up if your have comments, suggestions, additional interesting points, or even disagreements to share
  - I can learn from you too
  - New ideas and discussions make it much more lively and interesting
- Please be courteous of others
  - Turn off your cell phones

Today

- Introductory Material, Course Details, Review of CSE20
- Lab
  - Lab 1 assigned this week (1/21 – 1/27)
    - Knowledge test of CSE20
  - Lab 1 due next week (1/28 – 2/3)
- Reading Assignment
  - Review all Sections in Chapters 1 – 5 covered by CSE20 (Not graded)
  - Sections 6.1 – 6.5 (including participation activities)
    - Work on the Participation Activities in each section to receive participation grade at the end of semester (based on at least 80% completion)

Course Overview

- Prerequisites
  - CSE20
  - Basic knowledge of Computer Science
- CatCourses
  - Check regularly for announcements and lecture slides.
  - Labs & Project Assignments will be posted and submitted there.
  - Grades for assignments will also be found there (secure).
- 1 Lecture and 1 Lab per week
- 1 Mid-term exam (Mar 5, tentative)
- Final exam (May 10)
- 13 lab assignments (includes 2 exam review labs)
- 2 programming projects

Course Material

- Text Book: Programming in Java by Zyante
  - Sign up/in at zyBooks.com using your UC Merced Email ID
  - Enter zyBook code: **UCMERCEDCSE21ChandrasekharSpring2018** to subscribe (Case sensitive)
  - You will be asked to do some of the exercises in the text as part of your reading assignment (Graded!).
- You must subscribe your own copy. Participation grade will be evaluated based on the activities within the subscription account.
- If you had a subscription of the book from a prior semester and have the option to renew. In case of renewal avail the cheaper option (school term)

Grading

- Participation: 15%
- Projects: 15%
- Mid-term: 20%
- Lab assignments: 25%
- Final exam (comprehensive): 25%

- Grading (tentative):
  - > 88% scores at least an A-
  - > 73% scores at least a B-
  - > 59.5% scores at least a C-

Lab Rules

- Attendance is mandatory
  - Participation grade is evaluated from physical presence and observation of your working during lab hours.
- **Must show TA your lab before you can leave**
  - Easy to grade after since everyone gets it right.
  - Give you a chance to change your answers.
- Submit on CatCourses before the deadline (typically 1 week)
  - Grace period of THREE days after deadline.
  - No re-submission after grace period (exceptions may apply if approved by instructor before hand).
  - To ensure receiving full credit for all of your assignments, **verify the solutions with your TA or instructor before submissions**.

Project Rules

- 1 – 2 students per group
- All group members must submit their own solution in their CatCourses account
- Should be done outside of lab session hours unless you have completed the lab assignment already
- Same submission policy as labs, except for later deadline (typically 2 weeks)

Exams

- 45% of the course grade
  - Midterm 20%
  - Final 25%
- Open notes and open book (chapter printouts)
  - No electronic devices
- Practice Exams in Lab
  - For both midterm and final
  - Actual exam will follow the same format and order
  - Expect you to study hard so each problem will be harder on the actual exam

Hints for success

- Attend lecture
- Read the textbook and work on the activities
- Do & understand the assignments YOURSELF
- Create a portfolio to save all your work
- Take notes while reading and in lecture
- Ask questions: We are here to help you!
- Enjoy Learning!
- Helpful resources posted on CatCourses

Policies

- Don’t copy someone else’s code
- Don’t give your code away
- Don’t outsource your assignments
- Don’t use electronic devices in exams
- Don’t use electronic devices during lecture for purposes other than note taking
- Turn off speakers/cellphone during class

No Cheating!

- Communicating information to another student during examination.
- Knowingly allowing another student to copy one's work.
- Offering another person's work as one's own.
- I am serious!

About me

- Originally from India, moved to the US in 2002
- Pronounced: San-tosh Chun-druh-seh-kher
- Academic
  - Ph.D. from the University of Kentucky, Lexington in 2011
  - Postdoctoral Scholar at UCM from Aug 2012 till Sep 2016
  - Lecturer at UCM since Jan 2016
  - Research interests: Computer security and applied cryptography

Review of CSE 20

- Problem Statement
  - We want to survey the type of mobile OS students prefer. It will ask for a sample size and inquire for each person whether they like Android, iOS or both. It should report a breakdown of the data upon request.

The Program Skeleton

```
public class PreferenceMOS {

    public static void main(String[] args) {



    }

}
```

What's the first thing?

- Get input from user
  - How? Use a Scanner

Scanner

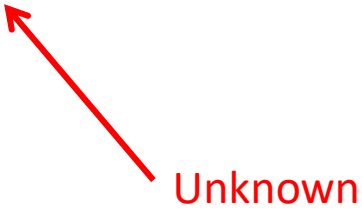
```
public class PreferenceMOS {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

    }

}
```



Import Class

```
import java.util.Scanner;

public class PreferenceMOS {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

    }

}
```

Steps

- Get input from user
  - How? Use a Scanner
  - What? To begin with, the sample size

Get sample size

```
import java.util.Scanner;
public class PreferenceMOS {
        public static void main(String[] args) {
                Scanner input = new Scanner(System.in);
                System.out.print("Enter the total number of students: ");
                int max = input.nextInt();
        }
}
```

Steps

- Get input from user
  - How? Use a Scanner
  - What? To begin with, the sample size
- Get samples

Get samples

- Ask to choose which one they prefer
  - Print
    - 1 for Android
    - 2 for iOS
    - 3 for Both
    - 4 for Other
- Use tally counters
  - if choice is
    - 1, android++
    - 2, ios++
    - 3, android++, ios++

Code to get a sample

```
System.out.println("Preference? Android (1), iOS (2), Both (3), or Other (4).");
System.out.print("Enter choice: ");
int choice = input.nextInt();

if (choice == 1)
        android++;
else if (choice == 2)
        ios++;
else if (choice == 3) {
        android++;
        ios++;
} else if (choice == 4)
        other++;
else
        System.out.println("Invalid choice.");
```

Putting it all together

```
import java.util.Scanner;

public class PreferenceMOS {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int android, ios, other;
        android = ios = other = 0;

        System.out.print("Enter the total number of students: ");
        int max = input.nextInt();

        System.out.println("Preference? Android (1), iOS (2), Both (3), or Other (4).");
        System.out.print("Enter choice: ");
         int choice = input.nextInt();`
        if (choice == 1) android++;
        else if (choice == 2) ios++;
        else if (choice == 3) {
                android++;
                ios++;
        } else if (choice == 4) other++;
        else System.out.println("Invalid choice.");

    }
}
```

Initialize all counters

Sample size

Get a sample

Steps

- Get input from user
  - How? Use a Scanner
  - What? To begin with, the sample size
- Get samples
  - For each person
    - Ask for choice (gather information)
    - Use tally counters

Repeat for each student

```
System.out.print("Enter choice: ");
choice = input.nextInt();
if (choice == 1) android++;
else if (choice == 2) ios++;
else if (choice == 3) {
        android++;
        ios++;
} else if (choice == 4) other++;
else System.out.println("Invalid choice.");

System.out.print("Enter choice: ");
choice = input.nextInt();
if (choice == 1) android++;
else if (choice == 2) ios++;
else if (choice == 3) {
        android++;
        ios++;
} else if (choice == 4) other++;
else System.out.println("Invalid choice.");
```

```
System.out.print("Enter choice: ");
choice = input.nextInt();
if (choice == 1) android++;
else if (choice == 2) ios++;
else if (choice == 3) {
        android++;
        ios++;
} else if (choice == 4) other++;
else System.out.println("Invalid choice.");

System.out.print("Enter choice: ");
choice = input.nextInt();
if (choice == 1) android++;
else if (choice == 2) ios++;
else if (choice == 3) {
        android++;
        ios++;
} else if (choice == 4) other++;
else System.out.println("Invalid choice.");
```

Too many people?

Looping

```
int choice = 0;
for (int i = 0; i < max; i++) {
        System.out.print("Enter choice: ");
        choice = input.nextInt();
        if (choice == 1)
                android++;
        else if (choice == 2)
                ios++;
        else if (choice == 3) {
                android++;
                ios++;
        } else
                other++;
}
```

Ignore invalid choices, and assume any other input is tallied under "other"

Steps

- Get input from user
  - How? Use a Scanner
  - What? To begin with, the sample size
- Get samples
  - For each person
    - Ask for choice (gather information)
    - Use tally counters
- Output

Output

```
System.out.print("See detailed count? Yes (1), or No (0): ");
int detailed = input.nextInt();
if (detailed == 1) {
        System.out.println("Prefer Android = " + android);
        System.out.println("Prefer iOS = " + ios);
        System.out.println("Prefer Other = " + other);
}
```

Final Code

```
import java.util.Scanner;

public class PreferenceMOS {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter the total number of students: ");
        int max = input.nextInt();
        int android, ios, other, choice;
        android = ios = other = choice = 0;
        System.out.println("Preference? Android (1), iOS (2), or Both (3).");

        for (int i = 0; i < max; i++) {
            System.out.print("Enter choice: ");
            choice = input.nextInt();
            if (choice == 1) android++;
            else if (choice == 2) ios++;
            else if (choice == 3) {
                android++;
                ios++;
            } else other++;
        }

        System.out.print("See detailed count? Yes (1), or No (0): ");
        int detailed = input.nextInt();
        if (detailed == 1) {
            System.out.println("Prefer Android = " + android);
            System.out.println("Prefer iOS = " + ios);
            System.out.println("Prefer Other = " + other);
        }
    }
}
```