 **Syllabus: Data Structures (CS 222), Jan-May 2016**

|  |  |
| --- | --- |
| **Instructor:**  Dr. Ayorkor Korsah  *Office*: Room 205-F  *Email*: [akorsah@ashesi.edu.gh](mailto:akorsah@ashesi.edu.gh)  *Office* hours: Mondays 2-4pm *or by appointment* | **Faculty Intern (Teaching Assistant)**  Esi Yeenuwa Yeboah  *Office*: Room 103 (Engineering building)  *Email*: [esi.yeenuwa@ashesi.edu.gh](mailto:esi.yeenuwa@ashesi.edu.gh)  *Office* hours: Tuesdays 2:30-4:30pm *or by appointment* |

**Meeting Times:**

Mondays 10:10 – 11:40am (R217)

Wednesdays 8:30am – 10:00am (Lab221)

Fridays 3:20pm – 4:50pm (Lab221)

**Course Overview:**

This course is about data structures; that is the methods of organizing large amounts of data. It is also about algorithm analysis; that is, the estimation of the running time of algorithms. Specifically, this course will cover fundamental abstract data types and their implementations as data structures, such as lists and trees, as well as asymptotic analyses of algorithms involving these data structures. Students will also learn about searching, sorting, graphs and algorithms on graphs.

**Course Objectives:**

Students will learn how to analyze algorithms before they are coded, to estimate their running time. They will also learn how to implement several different types of data structures. In all, by learning good programming and algorithm analysis skills simultaneously, they will be able to develop powerful and efficient software programs.

**Ashesi Learning Goals Addressed in this Course:**

1. **Critical Thinking and Quantitative Reasoning**: *An Ashesi student is able to apply critical thinking and quantitative reasoning to approach complex problems.* Students will develop the ability to choose appropriate data structures for given problems, to implement algorithms using those data structures, and to analyze the expected performance of the algorithms.
2. **Curious and Skilled**: *An Ashesi student is inquisitive and confident, has breadth of knowledge, and has attained a high level of mastery in their chosen field.* This course aims to develop skilled problem-solvers and programmers who have a widening understanding of the role of computer science and information systems in our complex world.
3. **Technology Competence**: *An Ashesi student is an effective and flexible user of technology*. This course focuses on developing a particular aspect of technology competence, namely implementing and using data structures, and analyzing the performance of algorithms. This will enable students to develop efficient, powerful, software programs.
4. **Leadership and Teamwork**: *An Ashesi student is adept at leading and functioning in teams*. Some assignments in this course require the students to work in teams of two students.

**Textbook:**

Michael T. Goodrich, Roberto Tamassia & Michael H. Goldwasser, *Data Structures and Algorithms in Java*.

**Other reference texts (available in the library)**

W. Savitch, Java: An Introduction to Problem-Solving and Programming.

Mark Allen Weiss, *Data Structures and Algorithm Analysis in Java*.

Robert Lafore, Data Structures & Algorithms in Java.

**Expectations:**

The instructor and faculty intern are committed to helping you to be successful in this course. In return, there are some fundamental expectations of you.

*Academic honesty*

You are expected to keep in mind at all times that *“An Ashesi student is an ethical, responsible and engaged member of his/her community”*. The work in this course is designed to help you develop knowledge and skills essential to your future career success. You can develop these skills only if you do the work yourself.

All the work that you turn in *must* be your own. In general, you are allowed and encouraged to brainstorm about problems with your peers. You can talk about ideas and approaches. However, you must write the program or the problem solution you turn in yourself. You must also acknowledge everyone that you discuss an assignment with in your submission. In code, this can be done with a comment at the beginning of the program. For most assignments, you will work individually. For some assignments, you may work in pairs. Each pair can collaborate on the code and hand in one submission, and the names of both partners must be in the submission. In your programs, if you happen to copy a section of code (e.g. a helper method) from any source for any reason you must include an appropriate citation in a comment above the code segment that you copied. You may not copy work from the internet, from your peers, or from any other source, without proper citation, nor should you enable others, either intentionally or by negligence, to copy your work. You may find that some problems we may work on may have solutions available on the internet. You may not, without

*Professionalism*

You are expected to interact with your course colleagues, as well as the instructor and teaching assistant in a professional and polite manner at all times.

*Participation*

Your active participation enriches the course experience for everyone. This includes asking and answering questions in class, and participating in any discussions. Do not be afraid to ask questions! Your questions will probably help others in the class as well.

**Evaluation Criteria**

*Your base grade in the course will be determined according to the following criteria:*

Class Preparation Assignments: 10%

Labs & Homework Assignments: 30%

Project: 10%

Tests 20%

Final exam: 30%

Class Preparation Assignments or CPAs, are short assignments based on assigned reading, which will be required to be completed almost daily, before class. In some cases, we will complete them in-class. We will spend the first 15 or so minutes of class discussing/grading these assignments. They will be graded on a scale of 0-2, where 0 represents no attempt, and 2 represents satisfactory completion. No make-ups are allowed for any reason (including absence due to illness), but the lowest two grades will be dropped.

Labs & homework assignments are longer assignments, usually involving programming and analysis, that you will work on weekly/bi-weekly. There will also be a small project with an accompanying report, in the second half of the semester. Further details will be forthcoming later in the semester.

There will be two tests in each half of the semester, and a final exam at the end of the semester. Note that a failing grade in the Tests and Final exam categories, may result in a failing grade in the course, regardless of your overall score. E.g. if you do excellently on the assignments, but fail the exams, your overall grade will be a failing grade. Thus, you must pay attention both to assignments and to exams.

Extra credit options on assignments: Some assignments may have some “extra credit”   
options for those who would like an additional challenge. Doing extra credit work can count as additional CPAs, replacing lower CPA scores if needed. Note that not doing extra credit work cannot harm you – you you not do extra credit work at the expense of the main assignment, because it is the grade on the main assignment that counts towards your base grade for the course.

**Late Policy**

Assignments are due as stated. Assignments handed in late will receive a 10% penalty per 24-hour period. As always, if there are mitigating circumstances (e.g. illness) for which an extension is needed, contact the instructor as early as possible ***before the assignment is due*** for the extension to be considered.

**Schedule**

Below is a tentative schedule. Updates will be posted on Courseware as the semester progresses.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Monday** | **Wednesday** | **Friday** |
| Wk 1  Jan 18 | Introduction | Java Review: Fundamentals  *Reading:* Skim 1.1 – 1.8 | Java Review: Program Design  *Reading:* 1.9 – 2.1 |
| Wk 2  Jan 25 | Java Review: Using Arrays  *Reading:* 3.1 | Java Review: Inheritance  *Reading:* 2.2 – 2.4  **Homework 1 due** | More Java: Casting, Generics & Nested Classes  *Reading:* 2.5 – 2.6 |
| Wk 3  Feb 1 | Linked Lists  *Reading:* 3.2 – 3.3 | Linked Lists  *Reading:* 3.4 – 3.6 | **Test 1** |
| Wk 4  Feb 8 | Algorithm Analysis  *Reading:* 4.1 – 4.3.2 | Algorithm Analysis  *Reading:* 4.3.3 – 4.4  **Homework 2 due** | Lab/Review/Discussion |
| Wk 5  Feb 15 | Recursion  *Reading:*  5.1-5.2 | Recursion  *Reading:*  5.3 – 5.4 | Recursion  *Reading:* 5.5 – 5.6 |
| Wk 6  Feb 22 | Stacks & Queues  *Reading:* 6.1 – 6.2 | Stacks, Queues & Lists  *Reading:*  6.3 – 7.2  **Homework 3 due** | Lists  *Reading:* 7.3 – 7.5 |
| Wk 7  Feb 29 | **Test 2** | Trees  *Reading:*  8.1 – 8.3 | Trees  *Reading:* 8.3 – 8.4 |
| Wk 8  Mar 7 | *MID-SEMESTER BREAK* | | |
| Wk 9  Mar 14 | Review/Discussion | Graphs  *Reading: 14.1 – 14.2*  **Homework 4 due** | Graphs  *Reading: 14.3* |
| Wk 10  Mar 21 | Priority Queues  *Reading:*  9.1 – 9.2 | Priority Queues  *Reading:* 9.3 – 9.4 | *No class (Easter)* |
| Wk 11  Mar 28 | *No class (Easter)* | Review/Discussion  **Project Submission 1 due** | **Test 3** |
| Wk 12  Apr 4 | Maps & Hash Tables  *Reading:* 10.1 – 10.2 | Hash Tables & Sets  *Reading:* 10.2 & 10.5 | Lab/Review/Discussion |
| Wk 13  Apr 11 | Sorted Maps & Binary Search Trees  *Reading:*  10.3 & 11.1 | Balanced Search Trees  *Reading:* 11.2 – 11.3  **Project Submission 2 due** | Lab/Review/Discussion |
| Wk 14  Apr 18 | Sorting  *Reading:* 12.1 – 12.3 | **Test 4** | Sorting  *Reading:* 12.3 – 12.4 |
| Wk 15  Apr 25 | Review/Discussion | Review/Discussion  **Homework 5 due** | *No class (Revision Period)* |
| Wk 16  May 3 | Final Exams | | |