

## **DEPARTMENT OF SCIENCE**

### **COURSE OUTLINE – WINTER 2018**

### CS 1150 (A3): Elementary Data Structures 3 (3-0-3) 90 Hours for 15 Weeks

<b>INSTRUCTOR:</b>	David Gregg	<b>PHONE:</b>	780-539-2976
<b>OFFICE:</b>	C-427	<b>E-MAIL:</b>	dgregg@gprc.ab.ca
<b>OFFICE HOURS:</b>	TBA		

#### **CALENDAR DESCRIPTION:**

The course provides a review of programming principles (specification, implementation and testing), and an extension of object-oriented concepts from CS1140 including data abstraction, modular program construction and program reuse. The emphasis is on dynamic data structures (eg. lists, string, stacks, queues, tables), and their associated algorithms (eg. recursion, traversal, sorting, searching, hashing).

#### PREREQUISITE(S)/COREQUISITE: CS1140

#### **REQUIRED TEXT/RESOURCE MATERIALS:**

Introduction to Java Programming by D. Liang. ISBN 10th Edition 0-13-376131-2. Please make good use of the on-line and library resources related to data structures also. See the CS1150 MOODLE page for additional materials.

#### DELIVERY MODE(S): In class lecture and lab.

#### **COURSE OBJECTIVES:**

This course introduces students to:

- object oriented analysis and design: specification, implementation and testing.
- fundamental data structures: strings, stacks, queues, lists, trees, heaps, sets, dictionaries and graphs.
- algorithms associated with data structures: recursion, traversal, sorting, searching, hashing.
- asymptotic complexity analysis of algorithms

### **LEARNING OUTCOMES:**

As a result of taking this course, students will gain the ability to:

- analyze problems, design algorithms and data structures to implement computational solutions to problems using an object oriented computer language.
- design and implement object oriented classes, using inheritance and polymorphism.
- design and implement array based and linked data structures like: strings, stacks, queues, lists, trees, heaps, sets, dictionaries and graphs.
- design and implement efficient accessor and mutator methods for data structures.
- describe and implement common algorithms related to searching, sorting, traversals, and hashing. (This includes recursive algorithms.)

#### **TRANSFERABILITY:**

University of Alberta University of Calgary \* University of Lethbridge \* Athabasca University Augustana Faculty, University of Alberta Concordia University College Canadian University College King's University College Grant MacEwan University \* An asterisk (\*) beside any transfer institution indicates important transfer information. Consult the Alberta Transfer Guide.

\*Warning: Although we strive to make the transferability information in this document up-to-date and accurate, the student has the final responsibility for ensuring the transferability of this course to Alberta Colleges and Universities. Please consult the Alberta Transfer Guide for more information. You may check to ensure the transferability of this course at Alberta Transfer Guide main page <a href="http://www.transferalberta.ca">http://www.transferalberta.ca</a> or, if you do not want to navigate through few links, at <a href="http://alis.alberta.ca/ps/tsp/ta/tbi/onlinesearch.html?SearchMode=S&step=2">http://step=2</a>

\*\* Grade of D or D+ may not be acceptable for transfer to other post-secondary institutions. **Students** are cautioned that it is their responsibility to contact the receiving institutions to ensure transferability

# **EVALUATIONS:**

Assignments	25%
Quiz	10%
Midterm Exam	25%
Final Exam	40%

# **GRADING CRITERIA:**

Please note that most universities will not accept your course for transfer credit **IF** your grade is **less than C-**.

Alpha Grade	4-point Equivalent	Percentage Guidelines	Alpha Grade	4-point Equivalent	Percentage Guidelines
A+	4.0	90-100	C+	2.3	67-69
А	4.0	85-89	С	2.0	63-66
A-	3.7	80-84	C-	1.7	60-62
B+	3.3	77-79	D+	1.3	55-59
В	3.0	73-76	D	1.0	50-54
B-	2.7	70-72	F	0.0	00-49

# COURSE SCHEDULE/TENTATIVE TIMELINE:

Sequence	Торіс		
1	Introduction and Review		
2	Strings and Files		
3	Thinking in Objects (Introduction to the Stack)		
4	Inheritance and Polymorphism		
5	Binary Search		
6	Abstract Classes, Interfaces, and Object Oriented Design		
7	Exception Handling		
8	Array Based Lists and Stacks		
9	Recursion		
10	Algorithm Analysis and Sorting $(n^2 \text{ and } n \log_2 n)$		
11	Linked Lists		
12	Stacks and Queues		
13	Trees		
14	The Heap		
15	Hashing		
16	Graphs		
17	Generics and Iterators		

### **STUDENT RESPONSIBILITIES:**

Assignments are to be handed in and/or demonstrated in the scheduled lab on the due-date. Late assignments will **not** be accepted. Students will be eligible for a passing grade, only if they obtain 37.5 out of a possible 75 marks (on exams).

## STATEMENT ON PLAGIARISM AND CHEATING:

Cheating and plagiarism will not be tolerated and there will be penalties. For a more precise definition of plagiarism and its consequences, refer to the Student Conduct section of the College Admission Guide at <u>http://www.gprc.ab.ca/programs/calendar/</u> or the College Policy on Student Misconduct: Plagiarism and Cheating at <u>http://www.gprc.ab.ca/about/administration/policies/</u>

\*\*Note: all Academic and Administrative policies are available on the same page.

# **Additional Information:**

CS 1150 A3	Instructor	Room	Day	Time
Lecture	David Gregg	J228	Tuesday, Thursday	11:30 to 12:50
Lab	David Gregg	J101	Wednesday	14:30 to 17:20