

William Paterson University
College of Science and Health - Department of Computer Science

Fall 2015 – Spring 2017 Assessment Cycle
Analysis of the Course Coverage and Assessment Report Data

Course Number: CS3420
Course Coordination Committee Members: Erh-Wen Hu (chair), John Najarian, Gilbert Ndjatou
Date: June 20, 2017

A. Course Prerequisites/Co-requisites

- a) **Problems/Issues Identified:**
No problem identified.
- b) **Suggestions for Improvement:**
N/A

B. Course Objectives

- a) **Problems/Issues Identified:**
No problem identified.
- b) **Suggestions for Improvement:**
N/A

C. Course Student Learning Outcomes

- a) **Problems/Issues Identified:**
No problem identified.
- b) **Suggestions for Improvement:**
N/A

D. Course Content

- a) **Problems/Issues Identified:**
No problem identified.
- b) **Suggestions for Improvement:**
N/A

E. Assessment of the CS Program's Student Outcome

Suggestions about S6: Demonstrate abilities to select appropriate data structures and to design algorithm to solve problems.

One major objective of this course is to introduce students to the different data structures that are often used to solve computer problems and the algorithms used to manipulate those data structures. Substantial among of exercises and tests are also used to test students' knowledge of these structures and their ability to understand and use the algorithms. This student outcome is therefore appropriate and well assessed.

F. Analysis of the Course Student Outcomes Assessment Data

The evaluations of the assessment data of the course student outcomes are listed in Section G below. In S2016, the outcome assessment are based on the numerical scores students received for 12 programming assignments, four tests, and a final. Similar set of programming assignments, tests and final exam were used in F2016, and S2017 semesters. Students have done consistently well on all outcomes during the three semester that these learning outcomes were assessed

G. Course Coverage and Assessment Report Data

S2016:

	Learning Outcomes	Where Measured	Percentage of Satisfactory Results*
1.	I am able to describe the concept of recursion, formulate basic recursive algorithms and implement them in C++.	Projects. tests & Final	>80%
2.	I am able to describe the differences between <i>C</i> and <i>C++ strings</i> and be able to program with both.	“	>85%
3.	I am familiar with <i>pointers and dynamic memories</i> and I am able use them for the implementation of various data structures including linked list, trees, and graphs	“	>90%
4.	I am familiar with various <i>searching and sorting algorithms</i> and am able to program and apply them in practical applications.	“	>80%
5.	I am able to describe the concept of <i>templates</i> and can program with both template functions and classes.	“	>90%
6.	I am able to describe the concept of <i>stack</i> , implement and apply it in practical applications.	“	>90%
7.	I am able to describe the concept of lists, implement and apply them in practical applications.	“	>80%
8.	I am able to describe the concept of <i>queues</i> , implement and apply them in practical applications	“	>80%
9.	I am able to describe the concept of <i>trees</i> , implement and apply them in practical applications.	“	>80%
10.	I am able to describe the concept of <i>set and map</i> , implement and apply them in practical applications.	“	>80%
11.	I am able to describe the concept of graphs, implement and apply them in practical applications.	“	>70%
12.	I am able to write practical programs using data structures and other feature defined in <i>C++ Standard Template Library</i> or STL.	“	>90%
13.	I am able to analyze algorithms and to estimate their efficiency using big-O notation.	“	>90%

F2016:

	Learning Outcomes	Where Measured	Percentage of Satisfactory Results*
1.	I am able to describe the concept of recursion, formulate basic recursive algorithms and implement them in C++.	Tests & projects	>60%
2.	I am able to describe the differences between <i>C</i> and <i>C++ strings</i> and be able to program with both.	“	>70%
3.	I am familiar with <i>pointers and dynamic memories</i> and I am able use them for the implementation of various data structures including linked list, trees, and graphs	“	>75%
4.	I am familiar with various <i>searching and sorting algorithms</i> and am able to program and apply them in practical applications.	“	>70%
5.	I am able to describe the concept of <i>templates</i> and can program with both template functions and classes.	“	>80%
6.	I am able to describe the concept of <i>stack</i> , implement and apply it in practical applications.	“	>80%
7.	I am able to describe the concept of lists, implement and apply them in practical applications.	“	>75%
8.	I am able to describe the concept of <i>queues</i> , implement and apply them in practical applications	“	>70%
9.	I am able to describe the concept of <i>trees</i> , implement and apply them in practical applications.	“	>65%
10.	I am able to describe the concept of <i>set and map</i> , implement and apply them in practical applications.	“	>60%
11.	I am able to describe the concept of graphs, implement and apply them in practical applications.	“	>60%
12.	I am able to write practical programs using data structures and other feature defined in <i>C++ Standard Template Library</i> or STL.	“	>70%
13.	I am able to analyze algorithms and to estimate their efficiency using big-O notation.	“	>70%

S2017:

	Learning Outcomes	Where Measured	Percentage of Satisfactory Results*
1.	I am able to describe the concept of recursion, formulate basic recursive algorithms and implement them in C++.	Tests & projects	>70%
2.	I am able to describe the differences between <i>C</i> and <i>C++ strings</i> and be able to program with both.	“	>80%
3.	I am familiar with <i>pointers and dynamic memories</i> and I am able use them for the implementation of various data structures including linked list, trees, and graphs	“	>85%
4.	I am familiar with various <i>searching and sorting algorithms</i> and am able to program and apply them in practical applications.	“	>80%
5.	I am able to describe the concept of <i>templates</i> and can program with both template functions and classes.	“	>85%
6.	I am able to describe the concept of <i>stack</i> , implement and apply	“	>85%

	it in practical applications.		
7.	I am able to describe the concept of lists, implement and apply them in practical applications.	“	>80%
8.	I am able to describe the concept of <i>queues</i> , implement and apply them in practical applications	“	>75%
9.	I am able to describe the concept of <i>trees</i> , implement and apply them in practical applications.	“	>75%
10.	I am able to describe the concept of <i>set and map</i> , implement and apply them in practical applications.	“	>70%
11.	I am able to describe the concept of graphs, implement and apply them in practical applications.	“	>65%
12.	I am able to write practical programs using data structures and other feature defined in <i>C++ Standard Template Library</i> or STL.	“	>80%
13.	I am able to analyze algorithms and to estimate their efficiency using big-O notation.	“	>75%