5/9/2026 DATE
☑ REQUIRED COURSE
☐ ELECTIVE COURSE

MSD DIVISION
☑ NEW COURSE
☐ REVISION

Lake Land College

Course Information Form

COURSE NUMBER:	MAT-165	**TITLE:** *(30 Characters Max)*	Computer Science II								
SEM CR HRS:	3.0	**Lecture:**	2.0	**Lab:**	2.0	**ICCB Lab:**	2.0	**ECH:**	4.0		
Course Level:	☐ Gen Ed/IAI ☐ Career/Technical ☑ Baccalaureate/Non-IAI ☐ Dev Ed/Not in Degree Audit			**Clinical Practicum:**	0.0	**Work-based Learning:**	0.0	**WBL ECH:**	0.0		
Course PCS & CIP:	11 - 11.0201	**IAI Code:**	CS 912	**Contact Hours (Minutes/Week)**							
Repeatable (Y/N):	N	**Pass/Fail (Y/N):**	N	**Variable Credit (Y/N):**	N	**Min:**		**Max:**		**16 Wks**	200	**8 Wks**	400
Prerequisites:	MAT-160										
Corequisites:	None										
Catalog Description: (40 Word Limit)	Covers principles of object-oriented programming and fundamental data structures. Students apply concepts, such as inheritance, polymorphism and abstraction while implementing linked lists, stacks, queues, trees and sorting/searching algorithms. Emphasis on program design, modular development, recursion and analysis of efficiency.										

List the Major Course Segments (Units)	Contact Lecture Hours	Contact Lab Hours	Clinical Practicum	Work-based Learning
Memory concepts and object fundamentals	2	2		
Inheritance and class hierarchies	4	4		
Abstract classes, interfaces and generics/templates	4	4		
Linked lists, stacks and queues	5	5		
Binary trees and recursion	5	5		
Searching, sorting and complexity	6	6		
Multi-file programming and integration	4	4		
TOTAL	30	30	0	0

EVALUATION							
QUIZZES	☐	**EXAMS**	☑	**ORAL PRES**	☐	**PAPERS**	☐
LAB WORK	☑	**PROJECTS**	☑	**COMP FINAL**	☑	**OTHER**	☐

COURSE MATERIALS	
TITLE:	Intro to Python for Computer Science and Data Science
AUTHOR:	Paul Deitel and Harvey Deitel
PUBLISHER:	Pearson Education
VOLUME/EDITION/URL:	1st edition
COPYRIGHT DATE:	2020

MAJOR COURSE SEGMENT	HOURS	LEARNING OUTCOMES
		The student will be able to:
Memory concepts and object fundamentals	4	1. Identify stack vs. heap memory. 2. Differentiate between instance/class variables and methods. 3. Apply object assignment and copy constructors.
Inheritance and class hierarchies	8	1. Explain inheritance and "is-a" design. 2. Design class hierarchies to model problems. 3. Apply polymorphism through virtual functions/dynamic binding.
Abstract classes, interfaces and generics/templates	8	1. Classify when to use abstract classes vs. interfaces. 2. Implement generics/templates for reusable code. 3. Compare abstraction and interface designs.

Linked lists, stacks and queues	10	1. Implement linked list insertion/deletion/traversal. 2. Use arrays and linked structures to implement stacks/queues. 3. Analyze efficiency of operations.
Binary trees and recursion	10	1. Implement binary tree insertion/deletion/traversal. 2. Apply recursive techniques to tree structures. 3. Evaluate recursion vs. iteration.
Searching, sorting and complexity	12	1. Implement searching and sorting algorithms. 2. Explain time/space complexity trade-offs. 3. Evaluate efficiency of algorithms.
Multi-file programming and integration	8	1. Organize programs into multiple source files. 2. Explain modular programming benefits. 3. Construct small projects with separated interface/implementation.
	60	

Outcomes*	Outcome Title	At the successful completion of this course, students will be able to:
Course Outcome 1	Obj-Orient Design	Design and implement programs using inheritance, polymorphism, abstraction and modular programming.
Course Outcome 2	Core Data Struct Alg	Construct linked lists, stacks, queues, trees and sorting/searching algorithms while evaluating their efficiency.
Course Outcome 3	Modular Effic Softw	Organize multi-file projects, apply recursion and use generics/templates to produce scalable and reusable code.
Primary Laker Learning Competency	Creative Thinking & Problem Solving: Students think creatively to solve problems.	
Secondary Laker Learning Competency	Quantitative Literacy: Students analyze data and mathematical patterns in real-life situations.	

**Course and program outcomes will be used in the software for outcomes assessment and should include at least 1 primary and 1 secondary Laker Learning Competency. Limit to 3-5.*