

**Oklahoma State University Institute of Technology**  
**Face-to-Face Common Syllabus**  
**Fall 2017**

**ITD 1253 Object-Oriented Programming Using C#**

Students learn how to design, code, and test applications in C# using object-oriented programming techniques. Topics include classes, data types, variables, methods, recursion, operators, control statements, inheritance and polymorphism, arrays, packages, interfaces, Input/Output, and strings.

**Course Purpose:**

This course teaches the student the basics of object oriented programming using C# and Visual Studio.

**Type of Course:** Theory/Lab

**Credit Hours:** 3; Total clock hours of theory per semester: 30;

Total clock hours of lab per semester: 45; Total clock hours of clinical per semester: n/a;

**Class Length:** Full Semester

**Class Days and Times:** M/W 9:30am – 11:50am

**Prerequisites:** ITD-1033

**Instructor Name:** Jim Strother

**Instructor Phone:** (918) 293-4798

**Office:** ET/IT 15E

**Instructor email:** james.strother@okstate.edu

**Contact:** The preferred method of contact is email. Please allow 24-48 hours to return correspondence during the normal work week.

**Instructor's Office Hours:**

Monday/Wednesday - 8:00 am to 9:30 am, 1:00pm to 3:30pm

Tuesday/Thursday – 8:00am to 11:30am, 1:00pm to 3:30pm

Other times available by appointment

**School Name:**

Information Technologies

**School Phone:**

(918) 293-5440

## **REQUIRED TEXT, REFERENCES, AND MATERIALS**

### **Texts:**

*Murach's C# 2015*, Joel Murach, ISBN 978-1-890774-94-3

### **References:**

Access to the Web site <http://www.murach.com>

Online Classroom

### **Materials:**

Portable Drive/USB Drive, pencil(s), notebook paper, broadband Internet access.

Visual Studio 2015

### **Uniform/Tools:**

none

<b>Estimated Cost for Text:</b>	\$58
<b>Estimated Cost for Materials:</b>	\$20
<b>Estimated Cost for Uniform/Tools:</b>	\$0
<b>Total Estimated Cost</b>	<b>\$76</b>

### **Upon completion of the course, students should:**

<b>COURSE OBJECTIVES</b>		<b>ASSESSMENT OF OBJECTIVE</b>
A.2	demonstrate proficiency in the use of a programming language to solve complex problems in a secure and robust manner	Programming Labs*
J.2	write simple and compound conditions within a programming language or similar environment (e.g., scripts, macros, SQL)	Programming Labs*
M.4	demonstrate knowledge of industry standard software development best practices	Programming Labs*

\*Aspects of the competency assessments may be used in the university's assessment of student learning.

## **COURSE ACTIVITIES**

In this course students will:

- Write C# classes and interfaces
- Create Object Oriented designs
- Implement Object Oriented designs in C#
- Document Object Oriented designs
- Use a C# Integrated Development Environment (IDE)
- Complete programs that illustrate Object Oriented Programming concepts and philosophies utilizing C#.
- Apply problem-solving skills to real-world problems.
- Participate in class discussions and class activities
- Participate in group and individual presentations.
- Take examinations and quizzes

## **EVALUATION - GRADES WILL BE BASED ON THE QUALITY AND COMPLETION**

**OF THESE TASKS:** *(NOTE-Please indicate the course specific evaluations. List assignment(s) used in the university's assessment of student learning as separate line items and marked with an asterisk.)*

Programming Assignments	35%
In-class activities – Review Questions	15%
Section Exams	30%
Digital Course Portfolio	5%
Professional Development**	5%
Final Exam*	<u>10%</u>
<b>TOTAL</b>	<b>100%</b>

<b>OSUIT Grading Scale</b>
A = 90%-100%
B = 80%-89%
C = 70%-79%
D = 60%-69%
F = 59% & below

\*The student's grade for this assignment will be used in the university's assessment of student learning. A 70% competency or higher receives a Pass rating. This Pass/Fail rating is independent of the student's course grade.

\*\* Information Technologies students are expected to participate in professional development activities as defined in the supplemental professional development activity guidelines in the online classroom.

Daily and/or weekly quizzes, small weekly assignments and similar type projects: Normal return time to student by next class meeting or no later than one (1) week.

Extensive assignments, large lab projects, extensive quizzes, exams and similar type projects: Normal return time to students in one (1) to two (2) weeks.

## **RECOMMENDED STUDENT COMPETENCIES/SKILLS**

Students will need to have successfully completed the prerequisite course.

## **AUTHORIZED TOOLS**

Students may use any/all course materials, including books and notes, while participating in classroom activities with the exception of-class exams. All quizzes, exams, and written assignments are to be completed independently; no collaboration with classmates is permitted and any instance of such will be considered academic dishonesty.

The class requires programs to be written and submitted as part of the grade. The programs are expected to be completed in C# only. Other programming languages are not acceptable for submission of assignments in this class.

## **LATE WORK POLICY**

Assignments must be completed and submitted on time. Absence or personal problems will not be considered an excuse for submitting assignments late. Late assignments are accepted only at the instructor's discretion and an adjustment may be assessed to the grade. Class demonstrations, workshops and/or training sessions will not be repeated. If a student is to be absent, it is the student's responsibility to make sure the work is submitted by the due date and time. To meet course competencies, students will be required to complete all assignments and course work.

## **TESTING**

### **Exams:**

Regular exams may include any combination of multiple choice, matching, true/false, fill in the blank, completion, hands on projects, programming assignments, essay and/or short answer questions. Exams may be taken early at the discretion of the instructor. Exams may not be taken late unless there is an accepted, excused, and documented absence.

Exams will be given on the date and time set by the instructor. Make-up exams will not be given without proper documentation. Make-up exams will only be given at times arranged with the instructor.

## **OTHER LAB AND CLASSROOM POLICIES**

**Each assignment must contain a heading.** Headings for assignments in this course include your name, due date, course ID, and instructor name. This heading is used for both electronic and hard copy work. Failure to use the correct heading will result in a reduction of points from the final assessment score of the assignment. Assignments submitted with no name will not be graded.

This is the heading table template.

Course Name	ITD 1253 – Object Oriented Programming with C#
Instructor	Jim Strother
Student Name	<student name here>
Due date	<assignment due date here>
Grade	<grade earned here>
Grading Comments	<instructor comments here>

Replace the material tags (stuff in <>) with the appropriate information

**SYLLABUS ATTACHMENT**

View the Syllabus Attachment, which contains other important information, by visiting [http://osuit.edu/center/student\\_syllabus\\_information](http://osuit.edu/center/student_syllabus_information)

Course Schedule		
Module	Topics and Competencies	Assignment(s)
1	<ul style="list-style-type: none"> <li>• Class Introduction</li> <li>• Class Orientation</li> <li>• Chapter 1: How to get started with Visual Studio</li> </ul>	Read Chapter 01 <ul style="list-style-type: none"> <li>• Chapter Assignments</li> <li>• Chapter Exercises</li> <li>• Chapter Programming Labs</li> </ul>
2	<ul style="list-style-type: none"> <li>• Chapter 2: How to design a Windows Forms application</li> </ul>	Read Chapter 02 <ul style="list-style-type: none"> <li>• Chapter Assignments</li> <li>• Chapter Exercises</li> <li>• Chapter Programming Labs</li> </ul>
3	<ul style="list-style-type: none"> <li>• Chapter 3: How to code and test a Windows Forms application</li> </ul>	Read Chapter 03 <ul style="list-style-type: none"> <li>• Chapter Assignments</li> <li>• Chapter Exercises</li> <li>• Chapter Programming Labs</li> </ul>
<b>End of Section 1</b>		
09/18 – 09/27	Section 1 Exam	Section 1 Exam Programming Portion
09/18 – 09/24	Section 1 Exam	Section 1 Exam Theory Portion
4	<ul style="list-style-type: none"> <li>• Chapter 4: How to work with numeric and string data</li> </ul>	Read Chapter 04 <ul style="list-style-type: none"> <li>• Chapter Assignments</li> <li>• Chapter Exercises</li> <li>• Chapter Programming Labs</li> <li>• Section 1 Exam</li> </ul>

Course Schedule		
Module	Topics and Competencies	Assignment(s)
5	<ul style="list-style-type: none"> <li>Chapter 5: How to code control structures</li> </ul>	Read Chapter 05 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>
6	<ul style="list-style-type: none"> <li>Chapter 6: How to code methods and event handlers</li> </ul>	Read Chapter 06 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>
7	<ul style="list-style-type: none"> <li>Chapter 7: How to handle exceptions and validate data</li> </ul>	Read Chapter 07 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>
<b>End of Section 2</b>		
10/16 – 10/25	Section 2 Exam	Section 2 Exam Programming Portion
10/16 – 10/22	Section 2 Exam	Section 2 Exam Theory Portion
8	<ul style="list-style-type: none"> <li>Chapter 8: How to work with arrays and collections</li> </ul>	Read Chapter 08 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> <li>Section 2 Exam</li> </ul>

Course Schedule		
Module	Topics and Competencies	Assignment(s)
9	<ul style="list-style-type: none"> <li>Chapter 9: How to work with dates and strings</li> </ul>	Read Chapter 09 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>
10	<ul style="list-style-type: none"> <li>Chapter 10: More skills for working with Windows forms and controls</li> </ul>	Read Chapter 10 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>
<b>End of Section 3</b>		
11/06-11/15	Section 3 Exam	Section 2 Exam Programming Portion
11/06-11/12	Section 3 Exam	Section 2 Exam Theory Portion
11	<ul style="list-style-type: none"> <li>Chapter 12: How to create and use classes</li> </ul>	Read Chapter 12 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> <li>Section 3 Exam</li> </ul>
12	<ul style="list-style-type: none"> <li>Chapter 13: How to work with indexers, delegates, events, and operators</li> </ul>	Read Chapter 13 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>



Course Schedule		
Module	Topics and Competencies	Assignment(s)
13	<ul style="list-style-type: none"> <li>Chapter 14: How to work with inheritance</li> </ul>	Read Chapter 14 <ul style="list-style-type: none"> <li>Chapter Assignments</li> <li>Chapter Exercises</li> <li>Chapter Programming Labs</li> </ul>

	Course Schedule	
Module	Topics and Competencies	Assignment(s)
<b>End of Section 4</b>		
12/04-12/10	Final Exam – Lab	<ul style="list-style-type: none"> <li>• Comprehensive Final Exam – Lab Programming Assignment</li> </ul>
12/04-12/10	Final Exam – Theory	<ul style="list-style-type: none"> <li>• Final Course Portfolio (due 8/20)</li> <li>• Comprehensive Final Exam – Theory Portion</li> <li>•</li> </ul>
12/14	Course Portfolio	Final course portfolio due: 12/14/2017
Refer to the drop boxes and chapter review question items in the online classroom for due dates on assigned class work *Schedule subject to change at instructor discretion and extenuating circumstances.		

Schedule is subject to change at instructor discretion.