

# EEL 3370 – C++ Programming for Embedded Systems Fall 2024 Page 1

Department of Electrical and Computer Engineering  
EEL 3370 – C++ Programming for Embedded Systems

**Instructor** : Dr. Herman Watson  
**Office Hours** : by Zoom meeting appointment  
Tuesday & Thursday 3:30 – 5:00 pm  
**Office** : EC - 3951  
**Sec. Phone** : 305.348.2807  
**Email** : watsonh\_fiu@yahoo.com (Note underscore) <<<<<<<< Student emails  
**Classroom/Time** : RVC: Online  
**Web Page** : <http://web.eng.fiu.edu/watsonh/>

## Catalog Description:

Object-oriented programming in C++ with emphasis on evaluation of alternative program design strategies applicable to Embedded Computing. Data Abstraction, Encapsulation, Inheritance, Polymorphism, Class design with data structures, Template Library, and wxWidgets applications library. These concepts work on big or small systems (3 Credits)

## References: : Open source materials are used as instruction materials

<b>Programming principles and practice using C++</b> <b>Bjarne Stroustrup</b> 978-0-321-54372-1	<b>C++ Primer</b> <b>Lippman, Lajoie, Moo</b> ISBN 0-321-71411-3
<b>Object-Oriented Programming in C++,</b> <b>Fourth Edition</b> <b>Robert Lafore</b> ISBN 0-672-32308-7	Cplusplus.com C++ Tutorial <a href="http://cplusplus.com/doc/tutorial/">http://cplusplus.com/doc/tutorial/</a>

## Course Objectives:

Through successful completion of the course, the student will:

- Understand and be able to analyze problem and develop an object-oriented solution.
- Confidently use C++ class syntax and semantics.
- Understand and be able to apply basic data structure concepts to real application.

**Relationship of course to program outcomes:**

- 1) an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics
- 2) an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors
- 4) an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts
- 6) an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions
- 7) an ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

**Topics Covered:**

- Introduction to C++ Programming
- Structures
- Class, Objects, and Strings
- Functions and Recursion
- Arrays and Vectors
- Pointers
- Overloading
- Encapsulation
- Inheritance
- Polymorphism
- Stream I/O
- Data Structures
- wxWidgets C++ library for Windows applications
- wxPong, wxTetris
- Data Structures
  - Linked Lists
  - Stacks
  - Queues
  - Trees

Grading Scale:		
A	92-100	"Florida International University is a community dedicated to generating and imparting knowledge through excellent teaching and research, the rigorous and respectful exchange of ideas, and community service. All students should respect the right of others to have an equitable opportunity to learn and honestly to demonstrate the quality of their learning. Therefore, all students are expected to adhere to a standard of academic conduct, which demonstrates respect for themselves, their fellow students, and the educational mission of the University. All students are deemed by the University to understand that if they are found responsible for academic misconduct, they will be subject to the Academic Misconduct procedures and sanctions, as outlined in the Student Handbook."
A-	90-92	
B+	88-90	
B	82-88	
B-	80-82	
C+	78-80	
C	70-78	
D	60-69	
F	< 60	

**Department Regulations Concerning Incomplete Grades**

*To qualify for an Incomplete, a student:*

1. Must contact (e.g., phone, email, etc.) the instructor or secretary before or during missed portion of class
2. Must be passing the course prior to that part of the course that is not completed
3. Must make up the incomplete work through the instructor of the course
4. Must see the Instructor. All missed work must be finished before last two weeks of the following term.

**Grading Scale: NOTE:** There is *no makeup exams* offered

Topic	Percentage
Exam 1 <i>no makeup</i>	20%
Exam 2 <i>no makeup</i>	20%
Exam 3 <i>no makeup</i>	25%
Homework	10%
Program of the Week	10%
DFS Project	10%
Participation	5%

**Policies:**

- 1) **Academic Misconduct:** For work submitted, it is expected that each student will submit their own original work. Any evidence of duplication, cheating or plagiarism will result in at least a failing grade for the course.
- 2) **Absences:** Resolution of absences and materials missed are student responsibility
  - 1) **Unexcused Absences:** Two unexcused absences are permitted during the term. More than two will result in the loss of points from your final grade. (1 point per absence above two, 3 points per absence above 5).
  - 2) **Excused Absences:** Only emergency medical situations or extenuating circumstances are excused with proper documentation.
    - 1) Review documentation with the lecturer,
    - 2) email as a written record to [watsonh\\_fiu@yahoo](mailto:watsonh_fiu@yahoo). (Note underscore)<<<<  
Name, SID, class, section, description and date of the absence
- 3) **On Time:** As in the workplace, on time arrival and preparation are required.
- 4) **Deadlines: Work is due before midnight on the date specified.** Late submissions and corrections within one week will receive up to half credit. After one week, **late work will not be accepted.** Late submissions are graded after the final exam. Participation deadlines are absolute – no late completions or makeups.
- 5) **Submissions: This class is paperless. Submissions are made using the web form listed on the class web site (online and in class sections).** All submissions must be on the class web site, not Canvas. All submissions must be
  - a) Captures of work must be whole screen images (include time and calendar)
  - b) Everything placed in a single ‘Word’ or ‘pdf’ document stored on your own cloud storage
  - c) Contain your name, date, and time of completion within the document
  - d) Permission: accessible by anyone with link and readable with a browser
  - e) Use a single URL link to view the document
- 6) **DO NOT** submit work by email.
- 7) Instructor reserves right to change course materials or dates as necessary.

**EEL 3370 – C++ Programming for Embedded Systems Fall 2024 Page 5**

<b>Mod</b>	<b>Date</b>	<b>3160 Weekly Topic Fall Semester</b>	<b>HW &amp; POW Due Tuesday</b>
1	08/26/24	Introduction to Bjarne Stroustrup and C++ V1, V2	<b>HW01</b> 09/03/24 Bjarne
2	09/02/24	Software Installation – IDE, Compiler, Application Library V3, V4 (09/02 Mon Labor Day)	<b>HW02</b> 09/10/24 wxHello
3	09/09/24	Functions, Pointers, Structures V5, V6	<b>HW03 PW3</b> 09/17/24 wxNotePad
4	09/16/24	Classes, constructors, destructors V7, V8	<b>HW04 PW4</b> 09/24/24 F.O.
5	09/23/24	Classes, overloading, pointers, this V9 <b>PQ1Classes</b> - Thursday	<b>HW05 PW5</b> 09/31/24 F. S.
6	09/30/24	Tue 10/01 Review V10 <b>Thur 10/03 Exam 1</b>	
7	10/07/24	STL & History V11 Iterators V12	<b>HW06 PW6</b> 10/15/24 Dialog
8	10/14/24	Iterators V13 Templates, Algorithms V14 <b>PQ2Iterator</b> - Thursday	<b>HW07</b> 10/22/24 <b>PW7</b> Data Structures
9	10/21/24	Templates, Containers V15	<b>HW08 PW8</b> 10/29/24 Template
10	10/28/24	Tue 10/29 Review V16 <b>Thur 10/31 Exam 2</b>	
11	11/04/24	WxWidgets – Tue: wxHelloWorld, V17 Thur: wxNotePad – wxSmith (10/31 Last Drop)	<b>HW09</b> 11/12/24 WxPong DFS-L1
12	11/11/24	Tue: wxPong V18 Thur: WxWidgets – wxTetris V20 <b>PQ3Pong</b> - Thursday	<b>HW10</b> 11/19/24 WxTetris DFS-L2
13	11/18/24	WxWidgets – Tue: wxSmithRenderTimer V19 Thur: Audacity, Gaming	DFS-L3 11/26/24
14	11/25/24	WxWidgets –Tue: Applications V22 (11/28-30 Thanksgiving)	<b>HW11</b> 12/03/23 DFS-L4
15	12/02/24	Tue 12/03 Review V23 <b>Thur 12/05 Exam 3</b> Senior Design Dav -12/06	
16	12/09/24	Final Exams Week – no final exam for this class	