Course Syllabus – CS 202 (Section 1001)
Spring 2019

1. Course: CS 202 - Computer Science II

2. Term Specific Information:
- Session: Spring 2019
- Credits: 3.0
- Class Interaction:
    Class Hours: Tu, Th, 4:30 pm - 5:45 pm
    Classroom: SLH 1
- Labs:

Lab Hours:

Section 1:	10:00am - 10:50am	@SEM 231C
Section 2:	11:00am - 11:50am	@SEM 231C
Section 3:	12:00pm - 12:50pm	@SEM 231C
Section 4:	7:00pm - 7:50pm	@SEM 231C
Section 5:	9:00am - 9:50am	@SEM 231C
Section 6:	6:00pm - 6:50pm	@SEM 231C
Section 7:	2:00pm - 2:50pm	@SEM 231C
Section 8:	1:00pm - 1:50pm	@SEM 231C

- Instructor: Christos Papachristos
    Office: ARF 004
    Email: cpapachristos@unr.edu, papachric@gmail.com
    Office Hours: Tu-Th 3:00-4:00 pm or by appointment
- Teaching Assistants:
    Email: xinyingw@, hsapkota@ , shuvo.k.paul@ , ycliu@ [nevada.unr.edu]

Office Hours:

Xinying Wang:	We 11am - 1pm	@ SEM 342D
Hemanta Sapkota:	Tu 11am-12pm / 1:30pm-2:30pm	@ SEM 342D
Shuvo Kumar Paul:	Mo 1 pm - 3 pm	@ ECC
Yuchuan Liu:	We 9 am – 11 am	@ SEM 342D

- NV PASS Leader: Kurtis Rodrigue
    Email: kurtisr@nevada.unr.edu
    Room: TBD
    Hours: TBD

3. Course Description:
   • Emphasis on problem solving and program development techniques. Typical numerical and non-numerical problems are examined. Design, implementation, and abstraction principles of elementary data structures.

4. Course Pre/Co-requisites:
   • CS 135 - Computer Science I with a "C" or better (C- and below does not count).
   • Corequisite(s): CS 105 or ENGR 100 for CSE Majors.

5. Recommended Preparation:
   • Basic Program Design, including the usage of the proper control structures for selection and iteration.
   • An understanding of, and the ability to use functions, including prototypes and definitions, along with passing parameters.
   • Declaration and usage of single and multidimensional arrays, including how to pass and use them as function parameters.
   • Basic Stream I/O, including the usage of external files.

6. Required Texts:
• No required textbook.
• Recommended: *Absolute C++ (6th Edition), by Walter Savitch*
                         C++ Primer (5th Edition),   by S. Lippman, J. Lajoie, and B. Moo

Some relevant online sources:

C++ (and C) Reference links for standard Libraries/Headers, Data Types, STL, etc. in latest and past standards:
   • cppreference.com
   • cplusplus.com

For the ones who want to remember the C language:
   • Essential C
   • An Introduction to C
   • An Introduction to the C Programming Language and Software Design
   • TutorialsPoint: Learn the C Programming Language

C++ related courses/links:
   • A Quick Introduction to C++
   • C++ Language Tutorial
   • TutorialsPoint: Learn the C++ Programming Language
   • An Introduction to C++
   • A Complete Guide to Programming in C++
   • Thinking in C++

You may find many other online sources. You may also use any C++ textbook as a supplement to the lecture.

7. Course Objectives:
   • Present the foundations for Object Oriented Design. Using the C++ programming language as a tool, Function and Operator Overloading will be covered and Templates will be introduced. The use of Pointers and Dynamic Memory will be covered leading to the introduction of Lists (array based and dynamically allocated linked lists), as well as coverage of other elementary data structures such as Stacks and Queues (array based and linked-list based). Basic algorithms including Recursion, as well as the Standard Template Library, to operate on these elementary data structures, will be covered.

8. Student Learning Outcomes:
Our graduates will have the ability to:
   • [UG -SLO2]   Design, implement, and evaluate a computing or engineering solution to meet a given set of requirements, with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.
   • [UG-SLO6]   Apply computer science theory and software development fundamentals to produce computing-based solutions.
   • [UG-SLO7]   Develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.

9. Course Topics Outline:
   • The following is not necessarily intended as a sequential ordering:

Topic	Lectures
C++ Primer, C++ I/O	2
Function Prototypes, Pointers, References	1
Structs	1
Classes	4
Inheritance	2
Polymorphism	2
Dynamic Memory	4
Classes and Dynamic Memory	1
Data Structures and Dynamic Memory	1
Arrays	2
Linked Lists	2
Stacks, Queues	2
Templates	2
Recursion	1
Midterm	1
Final Exam	1

10. Course Requirements:
• Lab Quiz assignments will comprise of problems that require a solution using a computer. We will be using the Linux boxes in the College of Engineering Computing. Answers will be given on sheet during the time of the lab section, and you will be required to hand in your answer sheet for grading at the end of the lab.
• Programming Project assignments will be given out weekly (or at most every two weeks). These programming assignments will be turned in through WebCampus (details will be provided by the course and lab instructors).
• All homework assignments (including exercises and projects) and all exams (quizzes, midterm and the final exam) are to be treated as individual and not collective efforts.
• All homework assignments should be considered as "open-book, take-home tests". If you need assistance with such an assignment, you may consult your instructor, TAs, PASS Leader, or a textbook. You may not receive substantive assistance in any from any other source (i.e., from another student, from computer center personnel, from paid or unpaid tutors, etc.). Any assistance you receive is to be documented in the comment section of your code.
• The only help you may receive from another student is with syntax errors or with questions regarding the computer system. Stealing another person's listing or having another person "ghost write" a lab will be considered cheating.

11. Exams:
• There will be 1 midterm exam and a comprehensive final exam. All exams will take place in the regular classroom.
• Midterm Exam – Thursday, March 14th, 4:30 pm - 5:45 pm (Tentative).
• Final Exam – Thursday, May 9th, 4:50 pm - 6:50 pm (Consult Online Schedule).

12. Grading Criteria, Scale, and Standards:
• The final grade will be based on:

Component	Percentage
Projects - Lab Quizzes/Assignments	50% (Breakdown: 40% - 10% each)
Midterm Exam	20%
Final Exam	30%

Average	Grade	Breakdown		
90% and above	A	A- : [90-95)		A : [95-100]
80-89.9 %	B	B- : [80-82.5)	B : [82.5-85)	B+ : [87.5-90)
70-79.9 %	C	C- : [70-72.5)	C : [72.5-75)	C+ : [77.5-80)
60-69.9 %	D	D- : [60-62.5)	D : [62.5-65)	D+ : [67.5-70)
59.9 % and below	F	F		

Plus/Minus grading will be assigned as indicated above. Students should consult the NSHE Grading Policy from the Board of Regents Handbook, and the Grades, Marks, and Grade Point Average policy from the UNR General Catalog.

• **Note:** You cannot earn a passing grade in the course without a passing grade on both the average of the midterm and final exams, as well as the average of the projects. Grade re-scaling may be assigned based on an outstanding or inferior final exam.

13. Late Work and Make-up Exams Policy:
• Each Lab Quiz assignment will be completed at the time of a Lab Section, and answer sheets will be collected at the end. Make-up Quizzes due to emergencies or other circumstances is only possible when in accordance to the University Absence policy (see UAM 3,020) and only after communication with the class instructor.
• Each Project assignment not turned in after its due date and time will be graded as late. The penalty for late submission will be as high as 20%. No Project assignment will be accepted 24 hours after the assigned deadline.
• Make-up Exams are only is possible when in accordance to the University Absence policy (see UAM 3,020) and only after communication with the class instructor.

14. Class Absence:
• Class presence will be required, tracked, and factored in, for the course's Lab Sections. For general university policy regarding class absence, see UAM 3,020.

15. Required Course Materials:
• You will be required to execute programming assignments in C++, and be able to provide functional executables and/or demonstrate actual program output. Hence, you will need access or possession of a PC with the required software suite, namely the Ubuntu Linux Operating System installed (minimal required version 14.04) and minimally the GNU Compiler Collection (minimal version 4.8). All mentioned software are Open-Source available.
• Most lectures will be provided as MS PowerPoint presentation files of PDF on WebCampus. You can view these with PowerPoint Viewer (free, not Open-Source) or another program of your choice (e.g. LibreOffice which ships with Ubuntu 14.04).
• Most assignments will be handed online on WebCampus. The file format will be PDF, and you may use Adobe AcrobatReader (free, not Open-Source) to view these (or *evince* which freely ships with Ubuntu Linux 14.04). You will need text editing software to hand in assignments, you may use any format of: PDF, DOC, DOCX, ODT. It is also mentioned that the LibreOffice suite shipping with Ubuntu 14.04 also includes text editing software.
• To submit assignments online over WebCampus you will need to archive your files. Any common archive type will be accepted (.zip, .zipx, .7z, .xz, .tar, .tar.gz, .tar.bz2, .rar). It is mentioned Ubuntu 14.04 freely ships with tar Archiving software.
• You will need network access and web browsing software. It is mentioned that Ubuntu 14.04 freely ships with Mozilla Firefox.

16. Unique Class Procedures / Structures:
• Some classes will potentially include video or other content from online sources. This will be projected in class for everyone, and the students will not be required to actively open the content at the time of the lecture.

17. Academic Standards:
• You should carefully read the [UNR Academic Standards Policy (UAM 6,502).](#) Your continued enrollment in this course implies that you have read it, and that you subscribe to the principles stated therein.
• In addition to the stated University standards, any assignment found to have more in common with another source (e.g., work of other students, online or published material, etc.) than is determined to be reasonable or acceptable by the course Instructor will be considered to be Academic Dishonesty.
• Per the University policy, the definition of Academic Dishonesty also applies to person(s) who provided the material(s) in question.
• When a student has demonstrated Academic Dishonesty, the policy of the Computer Science and Engineering Department is to apply the following minimum academic penalty of: 1) failure of the assignment with assigned grade of zero, and 2) a formal letter specifying the academic integrity breach and the associated sanction forwarded to the Office of Student Conduct to be placed in the student's permanent file.
• Depending on the egregiousness of the activity and for repeat offenders, sanctions beyond these minimums may be imposed at the discretion of the Instructor.

18. Disability Services:
• Any student with a disability needing academic adjustments or accommodations is requested to speak with the [Disability Resource Center](#) (Pennington Achievement Center Suite 230) as soon as possible to arrange for appropriate accommodations.
• This course may leverage 3rd party web/multimedia content, if you experience any issues accessing this content, please notify your instructor.

19. Audio and Video Recording:
• Surreptitious or covert video-taping of class or unauthorized audio recording of class is prohibited by law and by Board of Regents policy. This class may be videotaped or audio recorded only with the written permission of the instructor. In order to accommodate students with disabilities, some students may be given permission to record class lectures and discussions. Therefore, students should understand that their comments during class may be recorded.

20. Academic Dishonesty:
• The University Academic Standards Policy defines academic dishonesty, and mandates specific sanctions for violations. See the University Academic Standards policy: [UAM 6,502](#).

21. Academic Success Services:
• Your student fees cover usage of the [University Math Center](#) (775) 784-4433, [Tutoring Center](#) (775) 784-6801, and University [Writing Center](#) (775) 784-6030. These centers support your classroom learning; it is your responsibility to take advantage of their services. Keep in mind that seeking help outside of class is the sign of a responsible and successful student.

22. University Math Center:
• The [University Math Center](#) (UMC) is focused on helping students with mathematical and statistical concepts. While mathematics is used extensively in engineering, the UMC does not have the resources to help students with engineering courses. Engineering students are encouraged to use the UMC for help in their math classes, and they are welcome to use its computer lab and study area any time – regardless of course. However, UMC tutors cannot answer questions regarding engineering courses.

23. Safe Learning and Working Environment:
• The University of Nevada, Reno is committed to providing a safe learning and work environment for all. If you believe you have experienced discrimination, sexual harassment, sexual assault, domestic/dating violence, or stalking, whether on or off campus, or need information related to immigration concerns, please contact the University's Equal Opportunity & Title IX office at 775-784-1547. Resources and interim measures are available to assist you. For more information, please visit: https://www.unr.edu/equal-opportunity-title-ix.