Dr. John P. Abraham, Professor

Current Course Schedule (Spring 2014)

University of Texas – Pan American

Spring 2014

Professor: Dr. John P. Abraham. Office: Engineering Building Room 3.276
Web: faculty.utpa/jabraham Please check this site frequently for updates.
Email: jabraham@utpa.edu. Tele: 665-3550

ALL STUDENTS: PLEASE SUBMIT ASSIGNMENTS IN A MANILA FOLDER

CSCI 1201.09	**MW 11:45-12:35 ASB 2.110**		
CSCI/CMPE 1380 Comp Networks	**MWF 9:45-10:35 Eng 1.250**		
CSCI 6345.01	**Tu 5:45-8:25 Eng 1.268**		
Office Hours	**MWF 10:35-11:35 am Tu: 1-3pm**	**Please follow University Policies for dropping a class. No exceptions.**	**Jan 13: first day of classes Holidays: Jan 20, Apr 19 March 9-14 Spring Break Study days May 1-2 Final Exams May 3-9**

Follow me on Twitter: abrahamUTPA. You can ask me questions on Twitter.

TA: Nicholas Hinojosa <nhinojosa3@broncs.utpa.edu>

Dr. John P. Abraham

Course Description:

An introduction to computer science and computer programming is given, in which the fundamentals of a high-level programming language will be introduced. Methods of problem-solving, techniques of algorithmic development and concepts of structured object oriented programming will be emphasized.

Text and other Materials:

- Recommended: C++ Programming: Program Design Including Data Structures By D. S. Malik Edition: 5, illustrated Published by Course Technology, Cengage Learning, 2011 ISBN-13 978-0-538-79809-9, ISBN-10 0-538-79809-2. You may want to check www.bookfinder.com for used books or www.chegg.com for book rentals, among other sites. Local bookstores also rent this book.

Note: almost any C++ book should suffice (previous versions of this book or even books from other authors).

Course Topics:

This course will concentrate on techniques of problem solving and algorithmic design and will include lab experiences in the design and implementation of those algorithms in C++. C++ topics will include interactive input/output statements, file input and output statements, assignment statements, simple data types, selection and looping statements, functions, one and two dimensional arrays, user-defined data types, and structured data types. Time permitting, data abstraction and classes will be introduced.

Prerequisites:

Students are expected to have completed MATH 1340.

Course Objectives: After completing this course, the student should be able to:

Understand and carry out the steps involved in solving a problem using the computer, including the analysis, design, implementation, documentation and testing phases using methods of structured, modular and top-down style.

Be proficient with the programming environment and understand the system hardware, software and aspects of program translation.

Choose appropriate algorithms and data structures and proper syntax for problems involving selection control structures, various looping structures, procedures and functions, and for problems requiring array and other structures.

Learning Outcomes:

Upon completion of this course, students should be able to:

a. Be proficient with the programming environment and understand the basic aspects of program translation.
b. Analyze a programming problem and develop a solution algorithm.
c. Use the syntax and semantics of a higher-level language to implement their solutions to programming problems, including the correct use of:
    - simple types such as integer, character, floating-point variable types.
    - constants and variables declarations.
    - assignment, logical and arithmetic operators.
    - local and global variables.
    - selective control structures (if, nested if, switch).
    - iterative control structures (for, while).
    - functions (user-defined and predefined)
    - parameter passing to functions (by value and by reference).
    - arrays and other structures (such as records)
d. Document their solutions.
e. Apply simple testing and debugging strategies to identify and fix software faults.
f. Demonstrate proficiency in the use of logic and arithmetic operators and their precedence.
g. Use simple I/O to read and write character and numeric data to and from files, keyboard, and display.
h. Use simple sorting and searching algorithms.
i. Use standard documentation like online Help to determine the use of an unfamiliar tool.
p. Use teamwork roles and methods in the classroom

Course Grading Policies:

3 Tests	60% (20% per exam)
Attendance	10%
Lab Assignments	30%

Important: I do not assign extra credit work to let you modify your final grade. To ensure a good grade at the end of the semester make sure you get good ones in your assignments and tests.

Mondays and Wednesdays will be formal lectures and Fridays will be dedicated to lab exercises.

Your final grade will be based on the following scale:

A: 90-100% B: 80-89% C: 70-79% D: 60-69% F: 0-59%

Students are expected to:

- **Attend lectures and scheduled labs, and to study the text in preparation to contribute to discussions**. During labs students **MUST work on the current assignment**, no other activities will be allowed.
- **Have their electronic devices (cell phones, notebooks, music players, etc.) OFF at all times (tests, lectures and labs).**
- **Remain** in the classroom/lab during the **entire class period** unless **expressly** authorized by the instructor.
- Access their Blackboard account frequently to get information on course policies, assignments, tests, grades, etc. **All information posted on it will be assumed to be known by the student 24 hours later.**
- Access http://www.panam.edu/hop/ to read and adhere to the student code of conduct described there.

Make-up exams will **not** be given except by prior consent of the instructor. You **must** notify the instructor **within 24 hours** after missing the exam for determination of excuse. Examples of acceptable excuses would be death of an immediate family member, or illness, requiring physician's attention. Depending on the excuse, make-up exams may result in a loss of points. **All exams must be taken to be able to pass the course, missing anyone will result in an F as a final grade**.

Students who wish to **drop** the class should do it **on or before the** last day to officially drop or withdraw. After that drop withdrawal deadline, students will not be allowed to drop or withdraw from classes and will receive one of the following grades: **A, B, C, D, F, I, P, S,** or **U** for a class.

Students missing more than 10% of the classes will be reported to the Early Warning System program.

Note to students with disabilities:

If you have a documented disability which will make it difficult for you to carry out the work as I have outlined and/or if you need special accommodations/assistance due to a disability, please contact the Office of Services for Persons with Disabilities (OSPD), Emilia Ramirez-Schunior Hall, Room 1.101 immediately, or the Associate Director at MAUREEN@UTPA.EDU. Appropriate arrangements/accommodations can be arranged.

Verification of disability and processing of special services required, such as note takers, extended test time, separate accommodations for testing, will be determined by OSPD. Please do not assume adjustments/accommodations are impossible. Please consult with the Associate Director, OSPD, at extension 7005.

evaluation window closes at 11:59 pm on December 4th, the last day of classes for the fall semester. Students who complete their evaluations by December 4th will have priority access to their grades.

Lab Project Policies:

Unless stated otherwise, all projects are individual assignments and are expected to be the student's own work. The student may seek help with identifying syntax and runtime errors. General discussions regarding the solutions are encouraged, but giving or receiving major sections of code will be considered cheating and will be dealt with on an individual basis.

GUIDELINES FOR PROGRAMMING ASSIGNMENTS

ALL PROGRAMS AND OTHER ASSIGNMENTS ARE DUE AT THE BEGINNING OF THE CLASS PERIOD (5 minutes grace period). All assignments turned in after the class begins will be considered late (even if you come to class late).

Late Penalty: 5 minutes late to the end of the day -	10% penalty.
Next class period	20% penalty
Two class periods	30% penalty
One week+1 day	50% penalty

Will not accept after one week and one day.

Program Identification Section.

All programs should begin with a comment section that would include the following:

(IF YOU DO NOT INCLUDE ANY OF THESE YOU WILL BE PENALIZED 2 POINTS).

PROGRAMMERS NAME:________________________

CLASS AND SECTION:________________________ assignment #:________

DATE DUE:_____________ DATE TURNED IN:__________________

GENERAL DESCRIPTION OF THE PROGRAM (Program inputs, outputs and summary of what the program does).

Each function should be distinctly identified (block it off using asterisks). A brief description of its purpose should be given in comments.

Comments should be inserted as necessary throughout the program to convey the algorithm of the program. All programs must be tested thoroughly before submission. You should include program **runs** to indicate that every option of the program has been tested. **Programs turned in without any program run will receive a grade of zero.**

WHAT TO TURN IN:

1. Psuedocode, structure chart, flowchart, etc. as directed.

2. Program listing.

3. Program run.

Turn these in a folder.

GRADING:

Programs that do not run	0 points
Comments as explained above	15 points
Psuedocode, etc .	20 points
program listing	50 points
program run	15 points

Program correction and re-submission: up to 50% of missed points.