NASSAU COMMUNITY COLLEGE
DEPARTMENT OF MATHEMATICS/STATISTICS/COMPUTER PROCESSING
Course Outline for

CMP 217
C Programming Language

Curriculum	Interdisciplinary
Lab hours	None
Semesters offered	Indicated in Catalog
Length of semester	15 Weeks
Class hours	3
Credits	3
Text	The "C" Programming Language, by Kernigan, B. & Ritchie, D., Published by Prentice Hall, 2nd-l990

PREREQUSITE
Prerequisite: At least a C in CMP 211 or CMP 219 or CMP 218, or permission of the Department.

CATALOG DESCRIPTION

The C programming language is widely used in both the computer industry and as a tool for studying advanced topics in area of Computer Science. Topics include language syntax, scaler and compound data types, pointers, functions, operators, modular coding practices and files. Both the ANSI version and traditional C are discussed with emphasis on the former. This course is designed to meet the needs of the student considering further study in the area of Computer Science and Information Processing, and the computer science professional.

OBJECTIVES

General

At the end of the course, the student will be able to use C as a tool for developing computer program applications and study of advanced topics in Computer Science.

Specific

Study of the various ways which code can be written for both platform dependent and independent applications. Connection to other course work involving algorithms, programming languages and programming practices are stressed along with new topics. Relationship between the high level language feature and assembly language feature of C is studied. Important areas stressed are modular coding, functions and pointers.

Course Outline

Topics

	# Class hours
Overview of C	2
1.Program structure.	
2.Simple output. Use of putchar(), getchar(), printf().	
3.Use of preprocessor directives #include, #define.	
4.Flow of control if, while, for.	
5.Functions, definition, parameter passing mechanism.	
II.Lexical Elements, operators	3
1.Case sensitivity.	
2.Syntax rules, comments, keywords, identifiers.	
3.Operators. Both post and prefix increment operators.	
4.Assignment operator, bitwise operators.	
III.Fundamental Data types.	3
1.Declarations, expressions and assignments.	
2.Scaler Data types int, long, unsigned, short, char.	
3.Functions getchar() and putchar().	
4.Data conversion and casts of datatypes.	
IV.Flow of Control.	3
1.Relational, equality and logical operators.	
2.Short Circuit rules.	
3.Comma and conditional operator.	
4.Use of the switch statement.	
5.Branching techniques continue and break.	
V.Functions.	4
1.Function definition rules.	
2.Use of return statement.	
3.Prototyping.	
4.Scope rules, separate compilation of modules.	
5.Storage classes, auto, extern, register, static.	
VI. Arrays, pointers and strings.	6
1.One dimensional arrays. Initialization and usage as parameters.	
2.Pointers, address of (&) operator.	
3.Relationship between pointers, arrays and subscripts	
4.Pointer arithmetic.	
VII. Files.	2
1.C I/O functions fopen(), fgets() etc.	
2.UNIX I/O functions open(), read(), write().	
VIII. Records structure (struct).	2
IX. Linked-lists using C	
X.. Exams	2