

	Part II Description of Curriculum Change
1	
,	
	2. Summary of the proposed revision
	The principal revision is to change the programming language used in the
	course from FORTRAN to C++. The techniques and concepts taught, as well as the overall approach for the course, remains the same; only the language changes.
	There have been adjustments in the syllabus terminology to reflect the new
	language and adjustments in language processing to reflect the fact that C++ is to be taught on microcomputers rather than on the mainframe.
	3. Justification for the revision
!	FORTRAN is a programming language that is not being used as widely as
· · · · · · · · · · · · · · · · · · ·	
<u> </u>	

	I.	Catalog Description
		CO 110 Problem Solving and Structured Programming 3c-0l-3sh
.a.		(For science, mathematics, and computer science majors, and for others who have
	ì	
ent 19.000 Example		
<u> </u>		
		Ba.
F		
<u>'</u>		
	•	problem analysis and computer solution using flowcharting and the C++ language.
		<u></u>
.		

NEV	V CO:	110 Syllabus of Record	Attachment A
ш.	. Course Outline		
	A.	Introduction	2 hrs
		 History of computers Components of a computer Programming languages Compilation vs interpretation 	
	В.	The Programming Environment	4 hrs
		1. Editing	
1		· .	
	Ľ		

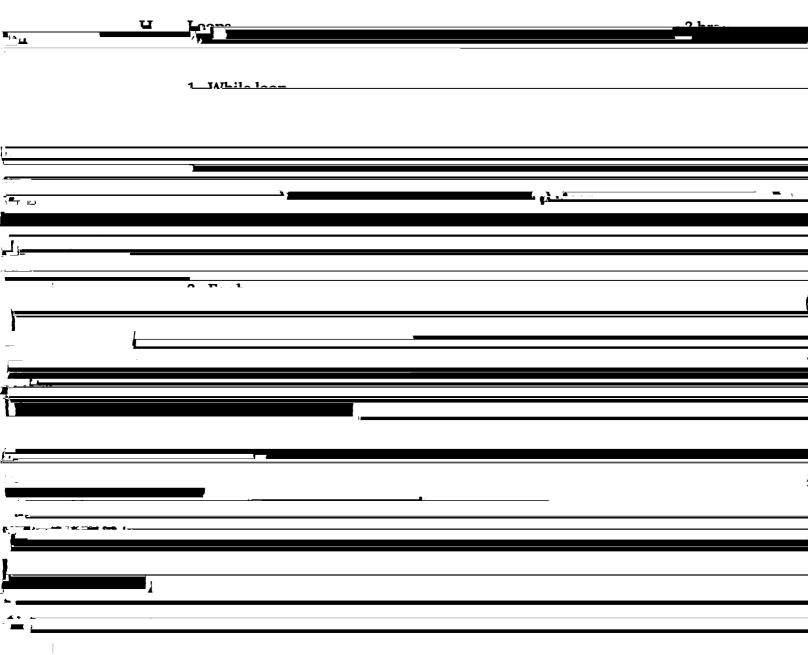
	\$ p - =	
	 -	
	5. Debugging6. Redirection of input and output	
C.	Algorithm development using flowcharts/psuedocode	4 hrs
	 Software engineering method Classic problems - maximum, minimum, sum, average 	
D.	Basic input and output	1 hr
E.	Data types	4 hrs
	 Constants Variables Expressions Library files and functions 	
F.	Simple Data Structures	7 hrs
	 One dimensional arrays Strings as arrays Multi-dimensional arrays 	

4. Records

NEW CO 110 Syllabus of Record

Attachment A

- 1. Single alternative
- 2. Double alternative
- 3. Multiple alternative
- 4. Nested structures



I. Programming language form

2 hrs

- 1. Syntax
- 2. Structured code
- 3. Documentation
- 4. Case sensitivity

primarily of multiple choice, true-false, and short answer questions.

30-35%

Programming assignments. There are approximately six programming assignments worth varying numbers of points that collectively count this portion of grade. Suggested tasks for the assignments include: linear search, sorting, 2-D array processing, interactive programming, simulation, sequential file processing, modularization.

10-15%

Class participation and quizzes. This may be based on written questions, verbal discussions, computer lab sessions, or other form of interaction.

Suggested Grading Scale:

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

V. Required textbook, supplemental books and readings

Adams, Joel, Leestma, Sanford, and Nyhoff, Larry. C++: An Introduction to Computing. Prentice Hall. 1995.

VI. Special resource requirements

None.

VII. Bibliography

Staugaard, Andrew C., Jr. <u>Structuring Techniques:</u> An <u>Introduction Using Turbo</u> C++.

Perry, Greg. <u>Turbo C++ Programming 101</u>. Sams Publishing. 1993.

I. Catalog Description

CO 110 Problem Solving and Structured Programming

3c-01-3sh

(For science, mathematics, and computer science majors, and for others who have a sufficiently quantitative orientation.) Basic structure of modern digital computers; batch processing vs. interactive time-shared online computing; problem analysis and computer solution using flowcharting and the FORTRAN language. Exemption or credit by examination possible.

II. Course Objectives

Upon successful completion of this course, the student will be able to

- A. Use the computing environment at IUP, including appropriate command language and editor(s).
- B. Develop algorithms from user problem statements.

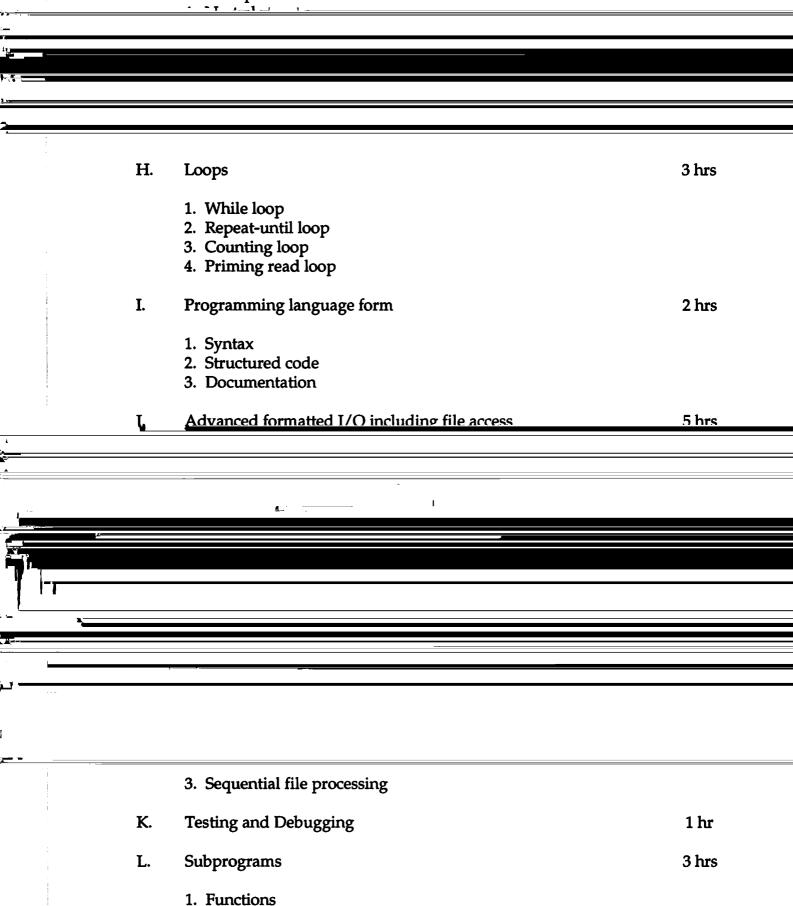
and/or pseudocode.

- D. Give commands to compile, link, and run their own programs, including using common options.
- E. Proficiently transform designs of problem solutions into programming language.
- F. Apply debugging and testing techniques to locate errors and determine the effectiveness of a program.

	OLD CO 110 Syllabus of Record Attachment B					
:	III.	Cour	se Outline			
:		A.	Introduction	2 hrs		
			1. History of computers			
			2. Components of a computer3. Programming languages			
			4. Compilation vs interpretation			
!		В.	The Computing Environment	6 hrs		
			1. Logging in, facilities			
			2. Editing			
			3. Command Language 4. Commission and linking with options			
			4. Compilation and linking, with options5. Command file and batch printouts			
:			6. Introduction to the debugger			
And the second second		C.	Algorithm development using flowcharts/psuedocode	4 hrs		
			1. Software engineering method			
			2. Classic problems - maximum, minimum, sum, average			
		_				
		D.	Basic input and output	1 hr		
		E.	Data types	3 hrs		
i			1. Constants			
			2. Variables			
			3. Expressions			
_			- -	- ·		
,						
i			•			
		F.	Arrays	6 hrs		
			•			
			1. On caling species at a			
_						
			2. Array I/O			
			2. Array 1/O 3. Multi-dimensional			
;			4. Classic problems - searching, sorting			
			1			

3. Multiple alternative

2. Subroutines



assignments include: linear search, sorting, 2-D array processing, interactive programming, simulation, sequential file processing, modularization. (180 points)

	<u>10 15∞</u>	Mas participation	and minns Thi	 Lace resultions	
1		Y			
<u> </u>					
- l					
•					
	<u> </u>				
x					
· -					
<u> </u>					
. (T				
<u>k</u>					
1 No. 1					
<u></u>					
<u></u>					
		F			
·		<u> </u>		 	
_					

To: James Wolfe, Chairman

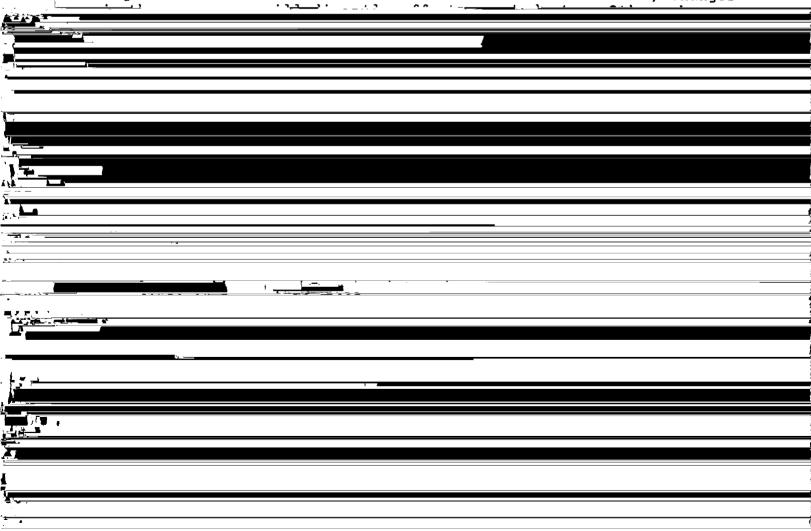
Computer Science Department Curriculum Committee

Gerald Buriok, Chairman Mathematics Department From:

Date: September 11, 1995

Subject: Computer Science Curriculum Changes

In your proposal of March 29, 1995, you outlined five changes the Computer Science Department wishes to implement in your curriculum. Since students majoring in our Mathematics and Applied Mathematics programs are required to take CO 110 Problem Solving and Structured Programming, and students in Applied Mathematics are required to take CO 250 Introduction to Numerical Methods, changes



Attachment C

Letters of Support

<pre>Erom:</pre>	GROVE:: WHITSON 7-SEP-1995 15:10:45.50
	JIM WOLFE GLBUTER Dual level courses in Computer s
Jim: The fol.	lowing is a memo that T amualso sending as a hard conveto your
•	
14.5	
-	* *
<u> </u>	Thursday Cantarhay 7 1005
<u></u>	
	•
	·— *

Chair Computer Science Dept. Curriculum Committee

From:

Dennis Whitson,

Chair, Physics Department

Re:

Use of the C language in CO 110 and CO 310

This issue was brought up at a Physics Department meeting on September 5, 1995. The Physics Faculty were in favor of switching to the use of the C language in CO 110 if the non-object approach was used. It was felt that the Physics Student needs to be introduced to the general subject of programming in a way that emphasizes algorithm development and fosters capabilities comparable to the current course which uses FORTRAN. You should leave object oriented programming to CO 310, though introducing it in the last few weeks of CO 110 would not be objectionable.

The use of FORTRAN in CO 250, while using C in CO 110 would inevietably degrade

Date: Thursday, September 7, 1995

To: Jim Wolfe,

14.4

Chair Computer Science Dent Curriculum Committee

Re: Use of the C language in CO 110 and CO 310

This issue was brought up at a Physics Department meeting on September 5, 1995. The Physics Faculty were in favor of switching to the use of the C language in CO 110 if the non-object approach was used. It was felt that the Physics Student needs to be introduced to the record to the second to be introduced to the second to