Università degli Studi della Basilicata
Scuola di Ingegneria

INSEGNAMENTO: Informatica			
DOCENTE: Russo Maria Grazia			
e-mail mariagrazia.russo@unibas.it			
Lingua di insegnamento	italiano		
n. CFU: 6	A.A.: 2013-14	sede: Potenza	Semestre: II

CONTENUTI

Architettura dei Sistemi Informatici: architettura dell'elaboratore; cenni alla rappresentazione binaria delle informazioni.

Ciclo di vita di un programma. Problema, algoritmo, programma. Storia dei linguaggi di programmazione. Il compilatore: funzione e processo di compilazione. Il collegatore: processo di collegamento. Esempi di compilatori e collegatori.

Introduzione alla programmazione C++: elementi di base; variabili; assegnazioni ed espressioni; tipi di dato semplici; strutture di controllo; operazioni di ingresso e uscita; gestione dei files.

Sottoprogrammi: procedure e funzioni; passaggio dei parametri. Metodologie di programmazione modulare.

Tipi di dato strutturati: array monodimensionali e bidimensionali, strutture. Soluzioni algoritmiche di base.

Elementi di base del Fortran.

Introduzione alla piattaforma di riferimento FreeMat per la programmazione MatLab. Elementi di base per la programmazione, sottoprogrammi, tipi strutturati, vettorizzazione.

METODI DIDATTICI

Lezioni frontali. Esercitazioni in sale attrezzate con postazioni informatiche (1 studente per computer).

TESTI DI RIFERIMENTO

- Materiale didattico fornito dal docente.
- John R. Hubbard -- Programmare in C++ (II Edizione, 2001) -- McGraw Hill Libri Italia - Milano

OBIETTIVI FORMATIVI

Gli obiettivi formativi relativi alla conoscenza e comprensione per questo corso prevedono che lo studente acquisisca le basi delle conoscenze della programmazione procedurale, anche in forma modulare, in diversi linguaggi, con particolare dettaglio relativo ai linguaggi C++ e Matlab.

Dal punto di vista delle capacità di applicare le conoscenze acquisite lo studente dovrà essere in grado di produrre codici in C++ e in Matlab per la risoluzione di problemi di semplice e media complessità, con particolare riguardo alla manipolazione di matrici e dati numerici.

PREREQUISITI

MODALITA' DI VERIFICA DELL'APPRENDIMENTO

Prova scritta a risposta multipla. Prova di programmazione al calcolatore.

PROGRAMMA ESTESO

Introduzione alla programmazione:

- elementi di base dei linguaggi procedurali (tipi, variabili, strutture di controllo, tipi strutturati, input/output, utilizzo di file ad accesso sequenziale)
- sintassi e semantica del linguaggio di programmazione **C++**
- tecniche algoritmiche di base sulle collezioni (somma, conteggio, massimi e minimi, verifica di condizioni, scambio e loro varianti)
- tecniche algoritmiche di base sulle matrici matematiche
- capacità di progettare e sviluppare soluzioni algoritmiche di varia complessità
- acquisizione dei dati da file a formato libero
- capacità di programmare in più linguaggi; capacità di applicazione dei concetti e delle tecniche previste dallo standard minimo **in linguaggio FORTRAN**
- conoscenza delle tecniche di test e verifica del codice.

Programmazione modulare:

- tecniche di programmazione modulare, progettazione e sviluppo dei sottoprogrammi e passaggio dei parametri, modello di esecuzione, utilizzo di librerie.

Matlab/Freemat:

- sintassi e semantica per la programmazione in linguaggio **MatLab**
- indicizzazioni e vettorizzazione;
- conoscenza delle principali funzioni Matlab relative all'Algebra di Matrici.

ALTRE INFORMAZIONI

Durante il corso lo studente può partecipare a prove intermedie per l'accertamento delle conoscenze.
Le prove sono di tipo scritto a risposta multipla.

Tutto il materiale didattico è fornito mediante apposito sito web dell'insegnamento.

Università degli Studi della Basilicata
Scuola di Ingegneria

COURSE: Computer Science			
TEACHER: Russo Maria Grazia			
e-mail mariagrazia.russo@unibas.it			
LANGUAGE	italian		
ECTS: 6 CFU	ACADEMIC YEAR: 2013-14	Campus: Potenza	Semester: II

TOPICS

Architecture of Information Systems: basic Architecture; binary representation of the information.

Life cycle of a program. Problem, algorithm, programs. History of programming languages. The compiler function and build process. The linker: linking process. Examples of compilers and linkers.

Introduction to C + + programming: basic elements, variables, assignments and expressions, simple data types, control structures, input and output operations, file management.

Subprograms: procedures and functions, parameter passing. Modular programming methodologies.

Structured data types: one and two dimensional arrays, records. Basic algorithmic solutions.

Basic Elements of Fortran.

Introduction to the platform for programming MatLab. Basic elements for programming, subroutines, structured types, vectorization.

TEACHING METHODS
Lectures. Laboratories.

TEXTBOOKS

- Educational material provided by the teacher.
- John R. Hubbard -- Programmare in C++ (II Edizione, 2001) -- McGraw Hill Libri Italia - Milano

LEARNING OUTCOMES

The educational objectives related to knowledge and understanding for this course require that the students acquire the basics of knowledge of procedural programming, even in a modular form, in several languages, with particular details relating to the languages C + + and Matlab.

From the point of view of the ability to apply the acquired knowledge the student will be able to produce code in C + + and Matlab for the resolution of problems of simple and medium complexity, particularly with regard to the manipulation of matrices and numerical data.

REQUIREMENTS

EVALUATION METHODS

Test with multiple choice answers. Test of programming on computers

DETAILED CONTENT

Introduction to Programming:

• basic elements of procedural languages (types, variables, control structures, structured types, input / output, use of sequentially accessed files)
• syntax and semantics of the programming language C++
• Basic algorithmic techniques on the collections (sum, count, maximum and minimum, conditional, exchange and their variants)
• algorithmic techniques on the basis of mathematical matrices
• Ability to design and develop algorithmic solutions of varying complexity
• acquisition of data from a file in free format
• Ability to program in multiple languages, the ability to apply the concepts and techniques provided by the minimum standard FORTRAN
• knowledge of the techniques for testing and verification of the code.

Modular programming:
• modular programming techniques, design and development of the sub-programs and parameter passing, execution model, use of libraries.

Matlab / FreeMat:
• syntax and semantics for programming in MatLab
• indexing and vectorization;
• knowledge of the main Matlab functions related to the Matrices Algebra.

FURTHER INFORMATION

During the course the student can participate in mid-term tests for the assessment of knowledge.
The tests are with multiple choice answers.

All teaching material is provided via the dedicated website of the course.