PROGRAMMA DEL CORSO DI ELEMENTI DI SISTEMI DI ELABORAZIONE E PROGRAMMAZIONE

SETTORE SCIENTIFICO

ING-INF/05

CFU

9

OBIETTIVI

Obiettivo dell'insegnamento è fornire gli elementi di base di conoscenza delle tecniche di programmazione con particolare riferimento ad algoritimi e paradigmi di programmazione e dei linguaggi di programmazione dedicati all'utilizzo dei Big Data.

Lo scopo è che l'allievo acquisisca le competenze per poter programmare ad oggetti, applicando coerentemente ed efficacemente gli strumenti messi a disposizione dai linguaggi imparati.

MODALITÀ DI ISCRIZIONE E DI GESTIONE DEI RAPPORTI CON GLI STUDENTI

L'iscrizione ed i rapporti con gli studenti sono gestiti mediante la piattaforma informatica che permette l'iscrizione ai corsi, la fruizione delle lezioni, la partecipazione a forum e tutoraggi, il download del materiale didattico e la comunicazione con il docente.

Un tutor assisterà gli studenti nello svolgimento di queste attività.

ATTIVITÀ DI DIDATTICA EROGATIVA (DE)

- 54 Videolezioni + 54 test di autovalutazione

Impegno totale stimato: 54 ore

ATTIVITÀ DI DIDATTICA INTERATTIVA (DI) ED E-TIVITY CON RELATIVO FEED-BACK AL SINGOLO STUDENTE DA PARTE DEL DOCENTE O DEL TUTOR

- Preparazione di un programma (in C, C++ o JAVA)
- Forum su Big data e strutture dati
- Svolgimento delle prove in itinere con feedback
- Svolgimento della simulazione del test finale

Totale 10 ore

ATTIVITÀ DI AUTOAPPRENDIMENTO

- 240 ore per lo studio individuale

LIBRO DI RIFERIMENTO

- Dispense del docente
- Testo di riferimento in via di definizione

MODALITÀ DI ESAME ED EVENTUALI VERIFICHE DI PROFITTO IN ITINERE

L’esame può essere sostenuto sia in forma scritta che in forma orale.

L’esame orale consiste in un colloquio nel corso del quale il docente formula di solito tre domande. L’esame scritto consiste nello svolgimento di un test con 30 domande. Per ogni domanda lo studente deve scegliere una di 4 possibili risposte. Solo una risposta è corretta.

Sia le domande orali che le domande scritte sono formulate per valutare sia il grado di comprensione delle nozioni teoriche sia la capacità di ragionare utilizzando tali nozioni. Le domande sulle nozioni teoriche consentiranno di valutare il livello di comprensione. Le domande che richiedono l’elaborazione di un ragionamento consentiranno di valutare il livello di competenza e l’autonomia di giudizio maturati dallo studente.

Le abilità di comunicazione e la capacità di apprendimento saranno valutate attraverso le interazioni dirette tra docente e studente che avranno luogo durante la fruizione del corso (videoconferenze, e-tivity report, studio di casi elaborati) proposti dal docente o dal tutor.

RISULTATI DI APPRENDIMENTO ATTESI

- Conoscenza e capacità di comprensione

Il corso intende fornire le conoscenze utili per comprendere metodi, tecniche e tecnologie per la progettazione e la implementazione di programmi. Esse includono: conoscenza e comprensione di fondamenti di problem solving, di algoritmi e strutture dati, di metodi e tecniche di astrazione; dei paradigmi di programmazione e dei linguaggi di programmazione, in particolare procedurali e ad oggetti; conoscenze dettagliate ed operative della programmazione ad oggetti. L’attenzione è rivolta essenzialmente ai metodi applicati in caso di uso di big data.

- Capacità di applicare conoscenza e comprensione

Il corso trasferisce la capacità di progettare e implementare programmi, anche partendo dalla progettazione di alto livello del sistema software a cui appartengono; nonché la capacità di usare librerie e repository di programmi, piattaforme tecnologiche e framework di programmazione; capacità di testing e debbugging di programmi.

- Autonomia di giudizio

Autonome capacità di giudizio sulla qualità dei programmi, sulla opportunità di riuso di programmi esistenti; nonché autonome capacità di giudizio, di valutazione comparativa e scelta di soluzioni, algoritmi e tools.

- Abilità comunicative

Capacità di comunicare in team di programmazione e con i progettisti di sistemi software; con fornitori di tecnologie e servizi per ambienti di programmazione; capacità di trasferire all'utente conoscenze e procedure per l'uso di programmi. Lo sviluppo di abilità comunicative, sia orali che scritte, sarà anche stimolato attraverso la didattica interattiva (con la redazione di elaborati da parte dello studente) e i momenti di videoconferenza attivati, ivi compreso la prova finale di esame.

PROGRAMMA DIDATTICO

1 - DAL PROBLEMA ALL'ALGORITMO

2 - DALL'ALGORITMO AL PROGRAMMA

3 - RAPPRESENTAZIONE E VERIFICA

4 - PROGRAMMAZIONE STRUTTURATA

5 - STRUMENTI PER LA PROGRAMMAZIONE

6 - IL LINGUAGGIO C

7 - VARIABILI E TIPI DI DATO

8 - OPERATORI

9 - VARIABILI E SELEZIONE

10 - ESERCIZI LINGUAGGIO C-BASE

11 - CICLO WHILE

12 - ITERAZIONE CON CONTROLLO

13 - ALTRI COSTRUTTI ITERATIVI

14 - FOR E SWITCH

15 - DO/WHILE E OPERATORI

16 - RIEPILOGHI ED ESEMPI

17 - PROGRAMMAZIONE MODULARE

18 - LE FUNZIONI

19 - UTILIZZO DELLE FUNZIONI

20 - ESEMPI DI FUNZIONI

21 - REGOLE DI VISIBILITA'

22 - LA RICORSIONE

23 - RICORSIONE E ITERAZIONE

24 - ARRAY NUMERICI

25 - ARRAY DI CARATTERI (STRINGHE)?

26 - VETTORI E FUNZIONI

27 - ORDINAMENTO DI ARRAY

28 - ALGORITMI DI RICERCA

29 - MATRICI

30 - PUNTATORI

31 - PUNTATORI E FUNZIONI

32 - ARITMETICA DEI PUNTATORI

33 - ARRAY-ESERCIZI

34 - CARATTERI E STRINGHE

35 - STRUTTURE DI DATI

36 - ELABORAZIONE DI FILE IN C

37 - IL PARADIGMA OBJECT ORIENTED

38 - PROGETTAZIONE E OGGETTI SOFTWARE

39 - IL LINGUAGGIO DI MODELLAZIONE UML

40 - DIAGRAMMA DEI CASI D'USO

41 - PROPRIETA' DEI CASI D'USO

42 - SOFTWARE OBJECT ORIENTED

43 - IL LINGUAGGIO C++

44 - INTRODUZIONE ALLE CLASSI IN C++

45 - CLASSI STRING E VECTOR

46 - GESTIONE ECCEZIONI, DEI FILES E DELL'EREDITARIETA' TRA CLASSI

47 - EREDITARIETA' E POLIMORFISMO

48 - IL LINGUAGGIO JAVA

49 - PROGRAMMARE IN JAVA

50 - MATEMATICA E VERIFICHE SU SEQUENZE DI DATI

51 - FILES, ARRAY E OOP IN JAVA

52 - IL LINGUAGGIO PYTHON

53 - LE BASI DEL LINGUAGGO PYTHON

54 - LINGUAGGIO PYTHON: FILE, CLASSI E NUMPY