UNIVERSITÀ DI PISA

FONDAMENTI DI PROGRAMMAZIONE CON LABORATORIO

CHIARA BODEI

Anno accademico	2019/20
CdS	MATEMATICA
Codice	017AA
CFU	9

Moduli	Settore/i	Tipo	Ore	Docente/i
FONDAMENTI DI PROGRAMMAZIONE	INF/01	LEZIONI	102	CHIARA BODEI NADIA PISANTI NICOLA PREZZA JACOPO SOLDANI

Obiettivi di apprendimento

Conoscenze

Lo studente che termina con successo il corso sarà in grado di dimostrare di conoscere i concetti basilari e i fondamenti dei linguaggi di programmazione. A questo scopo viene introdotto C come esempio di linguaggio di programmazione. Al termine del corso gli studenti devono essere in grado di tradurre un metodo astratto di risoluzione di un problema in un programma funzionante e di programmare in C ad un livello intermedio, elaborando algoritmi semplici.
Gli studenti devono inoltre dimostrare di aver acquisito una buona conoscenza di alcuni concetti relativi alla teoria dei linguaggi formali e alla teoria degli automi a stati finiti.

Modalità di verifica delle conoscenze

La prova scritta serve a valutare l'apprendimento delle nozioni teoriche sulla parte di linguaggi e le capacità di risoluzione algoritmica e di programmazione in C.
La prova pratica di programmazione ha l'obiettivo di accertare le capacità di programmazione di algoritmi in C, con particolare riferimento alla gestione di sequenze, array, liste ed alberi.
Con la prova orale, facoltativa, si verificheranno le conoscenze sui principali contenuti del corso, e la loro assimilazione, l'uso della terminologia appropriata e infine la capacità di affrontare problemi sia di tipo teorico che pratico legati ai contenuti del corso.

Capacità

Al termine del corso lo studente sarà in grado di:

- risovere algoritmicamente un problema e di programmarne la soluzione in C, tenendo conto anche dell'efficienza delle soluzioni proposte
- programmare in C a livello intemedio
- affrontare piccoli problemi ed esercizi sia di tipo teorico che pratico legati ai contenuti del corso

Modalità di verifica delle capacità

- Durante le prove scritte verranno richiesti esercizi per valutare l'apprendimento delle nozioni teoriche sulla parte di linguaggi e automi e di "problem solving" in C.
- Durante le prove pratiche di programmazione verranno richiesti esercizi per verificare la capacità di programmazione in C, con particolare riferimento alla gestione di sequenze array, liste ed alberi.
- Durante l'eventuale prova orale agli studenti verrà richiesto di saper presentare i principali contenuti del corso e di risolvere piccolo esercizi per dimostrare di saper utilizzare i principi appresi.

Comportamenti

Lo studente potrà acquisire e/o sviluppare la capacità di pensare in modo algoritmico alla soluzione dei problemi.

Modalità di verifica dei comportamenti

Durante le prove pratiche di laboratorio sarà anche valutata la modalità algoritmica di soluzione.

Prerequisiti (conoscenze iniziali)
Non sono richieste particolari conoscenze iniziali. Una conoscenza elementare della logica è certamente utile, così come una sufficiente familiarità con le tecniche di dimostrazione.

Indicazioni metodologiche
La frequenza è fortemente consigliata.

- Le lezioni frontali si svolgono con uso di slide e della lavagna.
- Le esercitazioni si svolgono in aula: gli studenti svolgono degli esercizi, anche in gruppo, sotto la supervisione del docente.
- Le esercitazioni pratiche di programmazione si svolgono in aula/laboratorio con i computer dell'aula e/o con quelli degli studenti, sfruttando un'apposita piattaforma di autovalutazione degli esercizi, che fornisce una verifica automatica della correttezza dei programmi caricati, con l'ausilio dei docenti e del personale di supporto in laboratorio
- L'interazione con il docente avviene con colloqui (in orario di ricevimento o su appuntamento) e tramite posta elettronica.
- Il materiale didattico, gli esercizi e alcuni esempi di testi di esame sono accessibile sulla pagina web del corso.
- Il corso prevede una prova pratica intermedia il cui superamento esonera lo studente dalla prova pratica di esame.

Metodi di insegnamento:

- Lezioni
- Esercitazioni
- Laboratori

Programma (contenuti dell'insegnamento)
I parte Linguaggio C, come esempio di linguaggio di programmazione.

- Introduzione all'informatica con brevi cenni all'hardware e alla rappresentazione binaria.
- Introduzione ai principi della programmazione, attraverso la presentazione dei costrutti linguistici di base e attraverso qualche cenno alla semantica operazionale di un semplice linguaggio imperativo.
- Primi elementi del linguaggio: variabili ed espressioni aritmetiche.
- L'istruzione if e gli operatori logici.
- Variabili di tipo char, int, long int, float, double
- Istruzione switch-case.
- Istruzioni for, break, continue.
- Istruzioni while e do-while.
- Array e matrici.
- Stringhe.
- Defi nizione di nuove funzioni.
- Passaggio dei parametri e prototipi. Il tipo void.
- Funzioni ricorsive e loro uso.
- Algoritmi di ordinamento.
- Considerazioni sulle prestazioni degli algoritmi e cenne alla loro complessità.
- Puntatori e allocazione dinamica della memoria.
- Tipi fondamentali e tipi strutturati.
- Liste concatenate.
- Alberi binari e alberi binari di ricerca con strutture concatenate.

II parte Cenni di teoria degli automi e dei linguaggi.

- Classi cazione delle grammatiche e dei linguaggi formali.
- Automi a stati finiti.
- Linguaggio riconosciuto da un automa a stati fi niti
- Automi deterministici e non deterministici e loro equivalenza.
- Pumping Lemma per i linguaggi regolari.
- Propriet a dei linguaggi regolari rispetto alle operazioni di unione, intersezione, complemento, prodotto e chiusura.
- Grammatiche libere dal contesto e linguaggi liberi.
- Pumping Lemma per i linguaggi liberi

Bibliografia e materiale didattico
Slides presentate durante le lezioni.
TESTI DI RIFERIMENTO

- S. Ceri, D. Mandrioli, l. Sbattella. Informatica: programmazione McGraw-Hill. In alternativa: D. Mandrioli, G. Cugola, L. Sbattella, P. Cremonesi e S. Ceri. Informatica: arte e mestiere

- J. Hopcroft, J. Ullman. Automi, linguaggi e calcolabilità. Pearson. Addison-Wesley.
- B. W. Kerninghan, D. M. Ritchie. Linguaggio C Pearson.
- Cormen, Leiserson, Rivest, Stein. Introduzione agli algoritmi e strutture dati. McGraw-Hill

BIBLIOGRAFIA DI APPROFONDIMENTO

- A. Bellini, A. Guidi. Linguaggio C, guida alla programmazione. McGraw-Hill.

Indicazioni per non frequentanti

Le modalità di esame per gli studenti non frequentanti sono identiche a quelle per gli studenti frequentanti.
Tutte le indicazioni su quanto fatto a lezione si trovano sulla pagina web del corso. Eventuale materiale sotto forma di lucidi o di esercizi è accessibile sempre a partire dalla pagina web del corso.

Modalità d'esame

L'esame è composto da:

- Prova scritta
- Prova pratica di programmazione in C
- Eventuale prova orale

La prova pratica consiste in un problema di programmazione da risolvere in circa due ore. Se superata, la prova rimane valida per tutti gli appelli dell'anno accademico.
Per accedere alla prova pratica e a quella orale occorre aver superato la prova scritta.
L'eventuale prova orale consiste in un colloquio tra il candidato e il docente, in cui vengono accertate le conoscenze del candidato sui principali argomenti del corso e le sue capacità nel risolvere problemi teorici e pratici.

Pagina web del corso

http://pages.di.unipi.it/bodei/CORSO_FP_19/FP/index.html

Ultimo aggiornamento 01/08/2019 13:12