UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Theory of Computation

2526-1-F1802Q151-F1802Q15102

Obiettivi

Conoscenza e comprensione

L'insegnamento di Teoria della Computazione ha l'obiettivo fondamentale di far acquisire allo studente gli strumenti teorici che permettono di comprendere la complessità computazionale dei problemi, come sono classificati in base alla loro complessità e quali sono le metodologie per risolverli, in modo esatto o con performance di approssimazione dimostrabili. Introduce anche metodologie algoritimiche e strutture dati avanzate per affrontare problemi fondamentali su testi di grandi dimensione.

In particolare lo studente sarà in grado di:

- Comprendere i concetti fondamentali del principale modello di calcolo (Macchina di Turing) e delle sue varianti
- Comprendere quali problemi possono essere risolti da un computer (decidibilità) e quali non possono essere risolti in un tempo ragionevole (intrattabilità)
- Comprendere gli algoritmi di approssimazione, cioè algoritmi che ottengono in tempi ragionevoli una soluzione dimostrabilmente vicina all'ottimo per problemi intrattabili in modo esatto
- Comprendere gli algoritmi parametrici, cioè algoritmi che permettono di affrontare in modo esatto alcuni problemi intrattabili limitando la crescita esponenziale dei tempi di calcolo dei loro algoritmi risolutivi a un parametro dell'istanza che risulta essere di valore limitato in casi pratici
- comprendere le basi teoriche e le applicazioni pratiche della ricerca esatta e approssimata di stringhe e delle strutture dati per l'indicizzazione di stringhe

Capacità di applicare conoscenza e comprensione

Lo studente sarà in grado di:

- capire come un problema può essere ridotto ad un altro per dimostrare la complessità relativa dei problemi
- classificare i problemi in base alla loro complessità computazionale (ad esempio, P, NP, NP-completi)

Autonomia di giudizio

Lo studente sarà in grado di:

- valutare se un problema è computabile o meno
- valutare quale algoritmo di ricerca utilizzare in un determinato contesto applicativo
- valutare quale struttura di indicizzazione utilizzare in un determinato contesto applicativo

Capacità comunicativa

- acquisizione del formalismo specifico alla disciplina della teoria della computazione

Capacità di apprendere

- acquisizione di capacità di astrarre rispetto a problemi computazionali affrontati nel corso

Contenuti sintetici

Nozioni di base di teoria della computazione: macchine di Turing, decidibilità e intrattabilità dei problemi computazionali. Riduzioni tra problemi computazionaii e classificazione dei problemi in funzione della complessità computazionale. Complessità di approssimazione e parametrica. Algoritmi di string matching esatto e approssimato. Strutture di indicizzazione.

Programma esteso

1. Nozioni di base di teoria della computazione:

- Il modello di calcolo della macchina di Turing e equivalenza con le sue varianti principali (macchine multinastro, alfabeto non binario, macchine non deterministiche)
- Relazioni tra linguaggi formali e problemi computazionali
- Limiti di computabilità: indecidibilità dell'Halting Problem

2. Trattabilità e intrattabilità, cioè classificazione dei problemi in funzione della complessità computazionale:

- Decidibilità in tempo polinomiale su macchine deterministiche (P, coP) e su macchine non deterministiche (NP, coNP)
- Equivalenza di decidibilità non deterministica in tempo polinomiale e verificabilità deterministica in tempo polinomiale
- Riduzioni polinomiali fra problemi di decisione
- NP-completezza del problema di Soddisfacibilità (SAT)
- Dimostrazioni di NP-hardness e NP-completezza

3. Complessità di approssimazione
4. Algoritmi parametrici
5. Algoritmi di string matching esatto

- Automa a stati finiti
- Algoritmo di Knuth-Morris-Pratt
- Algoritmo basato su paradigma shift-and

6. Algoritmi di string matching approssimato

- Algoritmo basato su paradigma shift-and

7. Strutture di indicizzazione di testi

- Suffix Array
- Burrows-Wheeler Transform
- FM-index

Prerequisiti

Nozioni di base di linguaggi formali. Nozioni di base di algoritmi e strutture dati.

Modalità didattica

Lezioni ed esercitazioni in aula.

Tutte le attività sono tenute in presenza e non vengono registrate né trasmesse in streaming.
L'insegnamento viene erogato in Italiano.

Le attività previste sono 28 lezioni da 2 ore svolte in modalità erogativa nella parte iniziale ed in modalità interattiva nella parte successiva.

Materiale didattico

Dispense e slides del docente.
Libro: Sipser, Michael. Introduzione alla teoria della computazione.

Periodo di erogazione dell'insegnamento

Primo semestre.

Modalità di verifica del profitto e valutazione

La verifica dell'apprendimento consiste di una prova scritta.

La prova scritta è basata su domande a risposta aperta relative alle nozioni e tecniche presentate nel corso e su esercizi che richiedono l'applicazione delle nozioni e delle tecniche suddette.

La prova scritta viene valutata in base alla correttezza e completezza delle risposte.

Sono previste due prove scritte in itinere.

Orario di ricevimento

Il ricevimento è su appuntamento da concordare con i docenti.

Sustainable Development Goals