



UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Didattica dell' Informatica

2627-2-F1802Q119

Obiettivi

Conoscenza e capacità di comprensione

Lo studente acquisisce conoscenze avanzate sui fondamenti epistemologici dell'informatica come disciplina scolastica, sulle principali teorie dell'apprendimento applicate all'insegnamento dell'informatica (costruttivismo, costruzionismo, peer instruction) e sui quadri curricolari nazionali e internazionali (Indicazioni Nazionali MIUR, CSTA Standards, DigComp).

Comprende la differenza tra informatica come scienza e uso degli strumenti digitali, e sa collocare il pensiero computazionale nel dibattito scientifico internazionale della CS Education.

Il corso fornisce le basi teoriche e gli strumenti pratici per l'insegnamento dell'informatica nella scuola. Partendo dai fondamenti della CS Education, si sviluppano competenze di progettazione curricolare, scelta di metodologie didattiche attive e valutazione degli apprendimenti, con riferimenti alle Indicazioni Nazionali e ai percorsi abilitanti per i docenti.

Conoscenza e capacità di comprensione applicate

Lo studente è in grado di applicare le conoscenze teoriche alla progettazione concreta di percorsi didattici per la scuola secondaria. Sa utilizzare framework operativi come il backward design (Wiggins & McTighe) e la tassonomia di Bloom rivista per l'informatica per costruire unità di apprendimento (UdA) coerenti e verificabili.

Sa scegliere e utilizzare strumenti e ambienti di apprendimento appropriati (attività unplugged, Scratch, Python, PRIMM, robotica) in funzione del livello scolastico e degli obiettivi didattici

Autonomia di giudizio

Lo studente è in grado di valutare criticamente metodologie didattiche, strumenti e risorse per l'insegnamento dell'informatica, argomentando le proprie scelte in relazione al contesto scolastico, al livello degli studenti e agli obiettivi curricolari.

Sa riconoscere e affrontare le misconcezioni tipiche nell'apprendimento della programmazione e delle strutture dati, adattando le strategie didattiche di conseguenza. Sviluppa capacità riflessiva sulla propria pratica attraverso i diari riflessivi e il feedback strutturato post micro-teaching.

Abilità comunicative

Lo studente sa comunicare contenuti informatici in modo chiaro, efficace e adeguato al livello del destinatario (studenti di scuola secondaria di primo e secondo grado). Attraverso il micro-teaching, dimostra la capacità di condurre una lezione strutturata, gestire i tempi, rispondere alle domande degli studenti e adattare la comunicazione al contesto.

Sa fornire e ricevere feedback strutturato tra pari, utilizzando griglie di osservazione condivise, e sa redigere relazioni di riflessione sulla propria pratica didattica.

Capacità di apprendimento

Lo studente sviluppa la capacità di aggiornarsi autonomamente sui temi della CS Education, disciplina in rapida evoluzione, attraverso la lettura critica di articoli di ricerca, la partecipazione a comunità professionali e l'osservazione sul campo (uscite in scuole secondarie).

Acquisisce gli strumenti concettuali e metodologici per proseguire la propria formazione professionale continua come docente di informatica, anche in relazione ai percorsi abilitanti all'insegnamento (TFA e percorsi di abilitazione D.Lgs. 36/2022).

Obiettivi

Il corso fornisce le basi teoriche e gli strumenti pratici per l'insegnamento dell'informatica nella scuola. Partendo dai fondamenti della CS Education, si sviluppano competenze di progettazione curricolare, scelta di metodologie didattiche attive e valutazione degli apprendimenti, con riferimenti alle Indicazioni Nazionali e ai percorsi abilitanti per i docenti.

Contenuti sintetici

Conoscenze e comprensione

- Fondamenti epistemologici dell'informatica come disciplina scolastica
- Principali teorie dell'apprendimento applicate all'informatica (costruttivismo, costruzionismo)
- Quadri curricolari nazionali e internazionali

Capacità e abilità

- Progettare unità didattiche con obiettivi, metodi e criteri di valutazione coerenti
- Scegliere e utilizzare strumenti e metodologie (unplugged, Scratch, Python, PRIMM) in base al livello
- Condurre micro-lezioni e ricevere feedback strutturato tra pari

Programma esteso

Informatica come disciplina scolastica

Cos'è l'informatica? Perché insegnarla? Storia e identità della disciplina, differenza tra informatica e uso degli

strumenti digitali, il pensiero computazionale e i suoi limiti come concetto ombrello.

Keywords: Epistemologia dell'informatica, Pensiero computazionale Informatica vs. digital literacy

Teorie dell'apprendimento e didattica attiva

Costruttivismo, costruzionismo (Papert), peer instruction (Mazur), flipped classroom. Misconcezioni tipiche degli studenti nell'apprendimento della programmazione e delle strutture dati.

Keywords: Papert & Logo, Misconcezioni in programmazione, Peer instruction, Flipped classroom, Pair programming, metodologie di insegnamento unplugged, metodologia PRIMM

Progettazione curricolare e UdA

Dal curriculum alla singola lezione: framework backward design (Wiggins & McTighe), tassonomia di Bloom rivista per l'informatica, scrittura di obiettivi operativi, scelta dei contenuti per ciclo scolastico.

Keywords: Backward design, Tassonomia di Bloom, Indicazioni Nazionali, UdA pratiche, Differenziazione

Strumenti e ambienti di apprendimento

Attività unplugged (CS Unplugged), ambienti block-based (Scratch, App Inventor), testo (Python), robotica educativa, intelligenza artificiale nella didattica. Criteri di scelta in base a età e obiettivi.

Keywords: CS Unplugged, Scratch, Python didattico, Robotica, AI in classe, Accessibilità digitale

Valutazione e sviluppo professionale

Valutazione formativa e sommativa in informatica, rubriche e portfolio, autovalutazione degli studenti. Micro-teaching: ogni studente conduce una mini-lezione (15 min) ricevendo feedback strutturato. Cenni alla ricerca in CS Education.

Keywords: Valutazione autentica, Rubriche, Micro-teaching, Feedback tra pari, CS Education research

Prerequisiti

Gradita, ma non indispensabile, conoscenza del linguaggio Python.

Modalità didattica

Struttura dei 5 moduli

La progressione va dal "perché" (identità della disciplina) al "come fare" (strumenti e micro-teaching), con la progettazione curricolare al centro come cerniera tra teoria e pratica.

Distribuzione teoria/pratica: i moduli 1-2 sono più teorici (ma con discussioni attive), mentre 3-5 spostano progressivamente il peso sul laboratorio, che rappresenta circa il 50% del monte ore complessivo.

Delle 52 ore complessive del corso, 20 sono svolte a distanza in modalità blended learning. Sono previste attività di scambio con docenti di scuola secondaria (secondo modalità da stabilire, possibilmente con uscite di osservazione sul campo) in modo da osservare e sperimentare i metodi di insegnamento dell'informatica in itinere.

Materiale didattico

Orit Hazzan, Noa Ragonis, Tami Lapidot. Guide to Teaching Computer Science - An activity based approach. Third edition, Springer Nature Switzerland, 2020, ISBN 978-3-030-39359-5.

Slide del corso.

Articoli suggeriti dal docente.

Altre risorse:

Guzdial, M. (2015). Learner-Centered Design of Computing Education. Morgan & Claypool.

Wing, J. M. (2006). Computational thinking. CACM, 49(3), 33–35.

Wiggins, G. & McTighe, J. (2005). Understanding by Design. ASCD.

CSTA K–12 CS Standards (2017, aggiornamento 2022). csteachers.org

CS Unplugged. csunplugged.org — attività unplugged pronte all'uso.

MIUR Indicazioni Nazionali per il Liceo Scientifico e gli Istituti Tecnici (Informatica).

Periodo di erogazione dell'insegnamento

Secondo semestre

Modalità di verifica del profitto e valutazione

Esame orale basato su un esercizio completo di progettazione didattica in informatica. La valutazione sarà così suddivisa:

- Progetto UdA (50%): Lo studente progetta un'unità di apprendimento completa su un argomento di informatica per la scuola secondaria, comprensiva di obiettivi operativi, sequenza delle attività, strumenti scelti e criteri di valutazione degli studenti. Il progetto viene discusso oralmente durante l'esame, dove il candidato motiva le scelte didattiche e risponde a domande critiche del docente.
- Micro-teaching (30%): Lo studente conduce in aula una mini-lezione di 15–20 minuti su un argomento a scelta, rivolta al docente che simula il ruolo di classe di scuola secondaria. Alla lezione segue una riflessione in cui il candidato analizza punti di forza e criticità della propria performance, tenendo conto del feedback ricevuto.
- Partecipazione e diari riflessivi (20%): Nel corso delle settimane di lezione, lo studente redige brevi diari riflessivi (circa mezza pagina) dopo le sessioni laboratoriali, documentando cosa ha appreso, cosa lo ha sorpreso e come collegherebbe i contenuti alla propria futura pratica didattica. La valutazione tiene conto sia della regolarità nella consegna sia della profondità della riflessione.

Orario di ricevimento

Su appuntamento

Sustainable Development Goals

ISTRUZIONE DI QUALITÀ
