

Programmazione dell'attività didattica di
Informatica
Classe III sez. B a.s. 2017-2018

Michele Nunziata
16 novembre 2017

1 Obiettivi Generali dell'Indirizzo di Studi

Analizzare, dimensionare, gestire piccoli sistemi per l'elaborazione, la trasmissione, l'acquisizione delle informazioni sia in forma analogica che digitale, in applicazioni di vario tipo ma particolarmente rivolte ai settori tecnico-industriali e scientifici. Le conoscenze dovranno essere tali da consentire una agevole partecipazione alla realizzazione di progetti più grandi e alla loro gestione.

1.1 Obiettivi specifici

Far acquisire agli allievi abilità tecniche e competenze operative, tali da sviluppare le capacità d'impostare problemi; mettere in grado l'allievo di verificare la correttezza di un procedimento risolutivo, mediante l'uso di macchine elettroniche programmabili; fornire le competenze indispensabili per l'utilizzazione di un moderno sistema di calcolo; dare metodologie d'organizzazione del lavoro, quali la documentazione, i piani di prova, etc.; far conoscere i campi applicativi per comprendere gli aspetti tecnici organizzativi, culturali e sociali dell'uso dell'informatica.

1.2 Competenze operative

- Capacità di utilizzo di un moderno sistema.

- Capacità d'impostare problemi.
- Capacità di verificare la bontà di una soluzione.
- Capacità di organizzare e documentare un lavoro.
- Conoscenza delle metodologie proprie dell'informatica.
- Conoscenza e capacità di analizzare criticamente aspetti tecnici, organizzativi, culturali e sociali dell'uso dell'informatica nei vari settori di applicazione

2 Argomenti da trattare

2.1 Modulo 1 - Concetti generali

L'introduzione al corso oltre che chiarire il campo di indagine mette in luce tutti gli sforzi fatti per il raggiungimento dello stato attuale.

Unità didattica 1

- L'elaboratore, l'automatismo e le informazioni.
- Algoritmi.
- La Macchina di Turing.
- Breve storia degli elaboratori.
- Legami con altre discipline.

Unità didattica 2

- Il modello di Von Neumann.
- Sistema Operativo.
- Linguaggi di programmazione.
- Text editor.

Laboratorio Una sessione di lavoro con il terminale Linux di Ubuntu usando il text editor Vim.

2.2 Modulo 2 - Il Paradigma imperativo

Questo modulo, costituisce la porta principale di ingresso alla programmazione e la base di lancio verso paradigmi di programmazione più recenti ed evoluti.

Unità didattica 1 - Problemi ed algoritmi

- L'approccio imperativo ai problemi.
- Definizione di problema, dati: costanti e variabili, risultati.
- Il concetto intuitivo di algoritmo e sue proprietà.
- La descrizione degli algoritmi: passi numerati, flow-chart, pseudocodifica.
- La documentazione.
- Il concetto di azione elementare e di istruzione.
- La simulazione dell'esecuzione di un algoritmo: la trace table.

Unità didattica 2 - Costrutti fondamentali

- La sequenza;
- La selezione: unaria, binaria e multipla.
- L'iterazione: preconditionata, postcondizionata ed enumerativa.
- Equivalenza e completezza delle strutture di controllo.
- La programmazione strutturata.

Laboratorio Realizzazione di semplici algoritmi implementati in Turtlescript utilizzando l'ambiente di sviluppo Kturtle.

2.3 Modulo 3 - Linguaggio C

La conoscenza del linguaggio C è di fondamentale importanza in quanto oltre ad addestrare ad una programmazione corretta, da utilissime indicazioni anche sull'utilizzo di altri linguaggi di tipo imperativo.

Unità didattica 1 - La sintassi del linguaggio

- Il lessico C.
- Alfabeto e parole riservate.

- Gli identificatori.
- I numeri e le costanti letterali.
- L'astrazione sui dati.
- Il concetto di tipo.
- La struttura di un programma C.
- Dichiarazione delle costanti, dei tipi e delle variabili.
- Assegnazione, Input/Output.
- Selezione, iterazione.
- Espressioni aritmetiche: regole di precedenza fra gli operatori aritmetici.

Unità didattica 2 -I tipi di dati

- Intero.
- Reale.
- Carattere.

Unità didattica 3 - I dati strutturati

- Il concetto di array.
- Array monodimensionali e bidimensionali.
- Gestione degli array.
- Problemi di ordinamento e di ricerca.
- Il tipo record (struct).
- Array di records.

Unità didattica 4 - La programmazione modulare

- Metodologia top-down e bottom-up.
- Ambiente e regole di visibilità.
- Le funzioni.
- Il passaggio dei parametri.

Laboratorio Realizzazione degli algoritmi di base nel linguaggio C in ambiente Linux.

3 Metodi

Le varie unità didattiche saranno realizzate, negli aspetti teorici, con lezioni frontali, nelle quali sarà dato ampio spazio al confronto ed alla valutazione critica. Sarà valorizzato ed incentivato il lavoro di gruppo soprattutto nelle applicazioni di laboratorio.

4 Mezzi e strumenti

Notevole importanza sarà data al lavoro di laboratorio inteso non solo come pura esercitazione pratica ma soprattutto luogo nel quale verificare la bontà delle soluzioni individuate nella parte teorica. Verrà utilizzato il Laboratorio open source e il linguaggio di riferimento sarà il C.

5 Verifica

Le verifiche saranno realizzate tramite questionari e colloqui individuali, prove scritte e prove di laboratorio. I questionari saranno svolti alla fine di ciascuna unità didattica e potranno essere seguiti da colloqui individuali ogni volta che se ne ravvisa la necessità. Le prove scritte avranno una cadenza mensile mentre le prove di laboratorio saranno relative agli aspetti salienti delle unità didattiche svolte.

6 Valutazione dei colloqui e dei questionari

6.1 Conoscenze

- Non conosce o conosce solo parzialmente e non sempre correttamente i temi proposti.
- Conosce in modo corretto ma puramente scolastico i temi proposti.
- Conosce ampiamente ed approfonditamente i temi proposti.

6.2 Competenze

- Non sa individuare i concetti chiave.
- Sa analizzare alcuni aspetti significativi.
- Sa analizzare i vari aspetti significativi e individua appropriate soluzioni.

6.3 Capacità

- Non sa esprimere giudizi personali né operare scelte proprie.
- Esprime giudizi e scelte adeguati ma non sempre adeguatamente motivati.
- Esprime giudizi e scelte adeguati ampiamente e criticamente motivati.

7 Valutazione delle prove scritte e di laboratorio

7.1 Congruenza con l'argomento proposto

- Traccia svolta parzialmente.
- Traccia svolta completamente.
- Traccia svolta completamente con soluzioni particolarmente efficienti.

7.2 Conoscenza delle strutture informative di base

- Non sa individuare né utilizzare le strutture di base.
- Le individua e le utilizza con difficoltà.
- Le individua e le utilizza con padronanza.

7.3 Capacità di risoluzione delle procedure richieste

- Non propone alcuna soluzione.
- Propone soluzioni complete o parziali ma documentate.
- Propone soluzioni complete ed efficaci adeguatamente documentate.

8 Scrutinio finale

Saranno ritenuti sufficienti gli allievi in possesso dei seguenti requisiti:

- Conoscenza precisa delle parti principali di un Elaboratore visto a blocchi funzionali.
- Conoscenza del linguaggio di programmazione utilizzato in laboratorio, tale da consentire una corretta traduzione di un procedimento espresso in forma algoritmica.
- Conoscenza degli algoritmi di base.
- Capacità di scrivere e rendere eseguibili programmi risolutivi di semplici problemi.