

PROGRAMMAZIONE DIDATTICA

Disciplina: **INFORMATICA**

Classe: **4** I.T.I.S corso Abacus Anno Scolastico 2016/2017

Finalità:

Il corso di Informatica ha come fine principale quello di mettere in grado lo studente di affrontare la soluzione di un problema, posto dalla richiesta di un ipotetico committente, scegliendo le metodologie e gli strumenti SW più idonei, ed inoltre fornirgli la formazione di base che gli consenta di seguire con una certa autonomia l'evoluzione delle tecnologie informatiche.

Obiettivi:

L'obiettivo della programmazione curricolare annuale è far acquisire all'alunno le competenze per affrontare progetti di una certa dimensione anche con la metodologia orientata agli oggetti. Deve acquisire una certa flessibilità che gli consenta di analizzare e risolvere problemi di varia natura utilizzando gli strumenti più idonei alla natura del problema.

Al termine della quarta classe gli allievi dovranno raggiungere i seguenti obiettivi:

1. Applicare il paradigma della programmazione orientata ad oggetti
2. Progettare e realizzare interfacce amichevoli
3. Individuare le caratteristiche dei nuovi linguaggi di programmazione imparandone rapidamente l'uso
4. Riconoscere in un linguaggio di programmazione le caratteristiche salienti
5. Scegliere per rappresentare e gestire le informazioni, il tipo di organizzazione più adatto
6. Documentare software

Tali conoscenze permetteranno agli allievi di affrontare nella classe finale argomenti sempre più complessi.

Contenuti:

Ripasso e approfondimenti sulla programmazione imperativa

- Strutture dati complesse: array, matrici, record
- Algoritmi di ordinamento. Problemi ricorsivi

La comunicazione con il computer:

- Il linguaggio HTML e i CSS
- Uso di un applicativo per la creazione e la gestione di siti web

La programmazione ad oggetti:

- Classi ed oggetti.
- Variabili locali e variabili istanza
- Attributi e metodi.
- Variabili e metodi di classe
- Interfacce con l'esterno, incapsulamento e information hiding
- Ereditarietà.
- Classi astratte.
- Gerarchie di oggetti
- Polimorfismo.
- Casting

Il linguaggio Java:

- Introduzione al linguaggio e all'ambiente di sviluppo JCreator
- Programmazione ad oggetti
- La grafica e le interfacce utente grafiche
- La gestione degli eventi

Strutture di dati:

- Introduzione alle strutture dati complesse
- Alcuni esempi di ADT (pila, coda, lista)
- Strutture dati non lineari (grafi e alberi)

Modalità di lavoro:

1. Spiegazioni in classe e/o in laboratorio.
2. Utilizzo del laboratorio di Informatica.
3. Studio e svolgimento di esercizi a casa.
4. Assegnazione di compiti con stesura di relazioni sotto forma di siti web.
5. Recupero periodico.
6. Attività di approfondimento e ricerca.

Strumenti di lavoro:

1. Lavagna.
2. Testi in adozione.
3. Manuali di programmazione.
4. Laboratorio di Informatica.
5. Fotocopie di materiale didattico vario (articoli di riviste, appunti, schemi, ecc.).

Tipologie di verifica:

1. Verifiche formative orali, scritte, pratiche (se necessarie, su singole lezioni ed argomenti).
2. Verifiche sommative orali, scritte, pratiche.
3. Verifiche di recupero orali, scritte, pratiche (se necessarie).
4. Relazioni scritte in laboratorio o a casa.

Numero di verifiche:

Primo periodo (trimestre) mediamente 2 prove di verifica sommativa di tipologia varia,
 Secondo periodo (pentamestre) mediamente 4 prove di verifica sommativa di tipologia varia +
 2 prove comuni di tipo 2.

Requisiti minimi per l'attribuzione della sufficienza:

Lo studente:

- possiede i contenuti minimi: strutture dati, file, ADT, programmazione a oggetti, basi del linguaggio Java
- applica le conoscenze acquisite per la soluzione di problemi noti (saper sviluppare programmi in C+ con uso di strutture dati, saper realizzare semplici programmi ad oggetti in Java, saper sviluppare interfacce grafiche in Java)
- si esprime in modo semplice ma con uso del linguaggio specifico della materia, cerca soluzioni anche precostituite.

N°	Modulo	Obiettivi	Tempi	Metodologie	Risorse	Valutazione
1	Gli oggetti	1, 3, 4	20 h	lezioni frontali	laboratorio	prove strutturate
2	Introduzione al linguaggio Java	1, 2, 3, 4	40 h	lavoro di gruppo attività di lab	laboratorio	progetti interrogazioni
3	Le classi	1, 2, 3, 4	40 h	lezioni frontali lavoro di gruppo attività di lab	laboratorio	interrogazioni prove strutturate, progetti
4	Liste	1, 5, 6	20 h	lezioni frontali lavoro di gruppo attività di lab	laboratorio	progetti prove strutturate
5	File	1, 5, 6	30 h	lezioni frontali lavoro di gruppo attività di lab	laboratorio	progetti interrogazioni
6	HTML e CSS	2, 3, 4	20 h	lezioni frontali lavoro di gruppo attività di lab	laboratorio	progetti prove strutturate

ESERCITAZIONI DI LABORATORIO

PERIODO	ESERCITAZIONI	OBIETTIVI
Settembre Ottobre Novembre	Sviluppo di applicativi completi in C con menu di scelta, funzioni, strutture dati complesse	Saper utilizzare C per problemi ricorsivi e strutture dati complesse
Dicembre Gennaio Febbraio	Sviluppo di pagine web con HTML e CSS Sviluppo di semplici programmi Java con gerarchie di classi. Sviluppo di programmi Java con gestione delle eccezioni	Saper progettare e costruire siti web utilizzando HTML, CSS, JavaScript. Saper applicare il paradigma della programmazione orientata agli oggetti a problemi di varia natura.
Marzo Aprile Maggio Giugno	Sviluppo di programmi Java con interfaccia grafica. Sviluppo di programmi Java con uso degli stream	Conoscere, progettare ed implementare strutture dati complesse con Java. Saper elaborare file ad organizzazione sequenziale. Progettare e costruire interfacce d'utente amichevoli

DEFINIZIONE DEI REQUISITI MINIMI PER L'ATTRIBUZIONE DELLA SUFFICIENZA

MATERIA: informatica CLASSE: 4 INDIRIZZO: informatica Abacus

CONTENUTI	CONOSCENZE MINIME	COMPETENZE MINIME
<ul style="list-style-type: none">• Programmazione a oggetti• Basi linguaggio Java	Conoscere le basi della programmazione orientata agli oggetti Conoscere le basi dell'linguaggio Java Conoscere gli elementi di base per la costruzione di semplici interfacce grafiche in Java	<ul style="list-style-type: none">• Applica le conoscenze acquisite per la soluzione di problemi noti• Si esprime in modo semplice ma con uso del linguaggio specifico della materia, cerca soluzioni anche precostituite
Html e CSS	Saper realizzare pagine in Html utilizzo di elementi base dei Css	