



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

Curricolo verticale della disciplina **INFORMATICA**

Percorso di studio: **Istituto Tecnico**

Indirizzo di studio: **Informatica e Telecomunicazioni**

SECONDO BIENNIO

QUARTO ANNO

MODULO	PREREQUISITI	CONOSCENZE	ABILITÀ	COMPETENZE	ATTIVITÀ DI LABORATORIO	CONNESSIONI INTERDISCIPLINARI	TEMPI
MODULO 1 IL LINGUAGGIO JAVA: SINTASSI BASE, STRUTTURE DI CONTROLLO E STRUTTURE DATI DI BASE	<ul style="list-style-type: none">Conoscenza degli elementi fondamentali di un linguaggio di programmazione	<ul style="list-style-type: none">Caratteristiche generali di Java: ambiente di programmazione e struttura dei programmiIdentificatori e parole chiaveVariabili e costantiI tipi di datoIl casting per la conversione di tipoGli operatoriCommenti e la documentazione (JavaDoc)Gestione input e output (classe Scanner)Strutture di controllo: sequenza, selezione e ripetizioneArray monodimensionali e bidimensionaliLe eccezioni	<ul style="list-style-type: none">Scrivere i programmi utilizzando in modo corretto la sintassi del linguaggioRiconoscere le diverse fasi del lavoro di programmazione per codificare e validare gli algoritmiDefinire le strutture per dati dello stesso tipoGestire le eccezioni	<ul style="list-style-type: none">Progettare la struttura generale di un programma JavaUtilizzare correttamente la sintassi del linguaggio e le strutture di controlloOrganizzare i dati in array	<ul style="list-style-type: none">Utilizzo di un ambiente di sviluppo Software (ad esempio NetBeans, Eclipse)Esercizi di codifica di diversi algoritmi in linguaggio Java	<ul style="list-style-type: none">Inglese: linguaggio tecnico	settembre - ottobre



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

MODULO	PREREQUISITI	CONOSCENZE	ABILITÀ	COMPETENZE	ATTIVITÀ DI LABORATORIO	CONNESSIONI INTERDISCIPLINARI	TEMPI
MODULO 2 CONCETTI DI BASE DELLA PROGRAMMAZIONE A OGGETTI	<ul style="list-style-type: none"> Conoscenza della sintassi di base del linguaggio Java 	<ul style="list-style-type: none"> Programmazione orientata agli oggetti Le classi: attributi, metodi e costruttori Gli oggetti (le istanze delle classi) Diagramma UML delle classi Dichiarazione e utilizzo di una classe Dichiarazione degli attributi I livelli di visibilità: public, protected e private La dichiarazione di un metodo La creazione di oggetti: l'importanza del metodo costruttore L'utilizzo degli oggetti: invocazione di un metodo da parte di un oggetto Uguaglianza tra oggetti: il metodo equals ed il costruttore di copia Attributi e metodi statici. Il mascheramento dell'informazione nelle classi: Information Hiding 	<ul style="list-style-type: none"> Acquisire gli elementi teorici del paradigma a oggetti (OOP) Comprendere il concetto di astrazione Acquisire il concetto di costruttore e distruttore Conoscere la metodologia di documentazione delle classi (UML) Applicare il concetto di astrazione per modellare le classi 	<ul style="list-style-type: none"> Definire una classe con attributi e metodi Applicare i concetti di incapsulamento e information hiding Rappresentare classi e oggetti mediante diagrammi UML 	<ul style="list-style-type: none"> Creazione dei diagrammi delle classi UML Creazione di programmi in grado di usare classi progettate e codificate dagli studenti 	<ul style="list-style-type: none"> Inglese: linguaggio tecnico Esercizi che partano da un contesto suggerito da argomenti affrontati nelle altre discipline 	<p>novembre - dicembre</p>
MODULO 3 VETTORI DI OGGETTI, EREDITARIETÀ E POLIMORFISMO, STRINGHE	<ul style="list-style-type: none"> Modulo 1 e modulo 2 	<ul style="list-style-type: none"> I vettori di oggetti. L'ereditarietà: concetti di superclasse e sottoclasse Il grafo di gerarchia e la classe Object Ereditarietà singola e multipla I vantaggi dell'ereditarietà Dichiarazione e utilizzo di sottoclassi Il polimorfismo: overriding e overloading La parola chiave final Le stringhe e i suoi metodi: charAt, equals, length, substring, toLowerCase, toUpperCase 	<ul style="list-style-type: none"> Comprendere le differenze tra overloading e overriding Usare la progettazione orientata agli oggetti per programmi complessi. Individuare la specializzazione e la generalizzazione di una classe. Applicare i concetti di ereditarietà e polimorfismo. Definire gerarchie di classi. Saper creare e manipolare le stringhe in Java 	<ul style="list-style-type: none"> Classificare le classi e le relazioni tra di esse Applicare i concetti di incapsulamento e information hiding Riconoscere la gerarchia delle classi Rappresentare la gerarchia di classi e oggetti mediante diagrammi UML 	<ul style="list-style-type: none"> Creazione dei diagrammi delle classi UML Esercitazioni di codifica sugli argomenti affrontati nel modulo 	<ul style="list-style-type: none"> Inglese: linguaggio tecnico Esercizi che partano da un contesto suggerito da argomenti affrontati nelle altre discipline 	<p>gennaio - febbraio</p>



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

MODULO	PREREQUISITI	CONOSCENZE	ABILITÀ	COMPETENZE	ATTIVITÀ DI LABORATORIO	CONNESSIONI INTERDISCIPLINARI	TEMPI
MODULO 4 LA GESTIONE DEGLI EVENTI E LE INTERFACCE GRAFICHE	<ul style="list-style-type: none"> Modulo 1, 2 e 3 	<ul style="list-style-type: none"> L'interfaccia utente e i suoi elementi Le librerie grafiche AWT e Swing I controlli: pulsanti di comando, etichetta di testo, casella di testo, pulsante di opzione, casella di controllo e casella combinata La gestione degli eventi Le finestre di dialogo e i menu 	<ul style="list-style-type: none"> Individuare le diverse tipologie di eventi Comprendere il ruolo degli elementi grafici Capire la struttura delle classi AWT e Swing Comprendere il ruolo degli ascoltatori di eventi Utilizzare i controlli e gli oggetti 	<ul style="list-style-type: none"> Realizzare progetti con componenti grafici GUI Distinguere tra classi e oggetti, controlli ed eventi Scegliere le tipologie di controllo adatte alle attività da svolgere Gestire gli eventi sulle finestre e sui componenti 	<ul style="list-style-type: none"> Realizzazione di progetti con interfaccia grafica in Java 	<ul style="list-style-type: none"> Inglese: linguaggio tecnico Esercizi che partano da argomenti affrontati nelle altre discipline 	marzo - aprile
MODULO 5 APPLICAZIONI WEB BASED: I FORM HTML ED IL LINGUAGGIO JAVASCRIPT	<ul style="list-style-type: none"> Html di base affrontata nel programma di terzo anno nella materia di sistemi e reti 	<ul style="list-style-type: none"> Ripasso dei principali elementi di HTML: struttura di una pagina, paragrafi, elenchi, tabelle link e immagini I form Il tag input e i suoi tipi: text, password, email, url, tel, date, range, radio, checkbox, button, reset, submit Il tag textarea Il tag select: il menù a tendina Il linguaggio JavaScript Differenza tra DOM e BOM Accedere e manipolare gli elementi della finestra del browser L'oggetto document I moduli e la convalida con JavaScript: getElementById, getElementsByTagName e query.Selector Il meccanismo degli eventi 	<ul style="list-style-type: none"> Saper creare dei form con i principali tag di tipo input ed i menu a tendina Comprendere il ruolo degli script JavaScript Comprendere l'interazione tra moduli HTML e JavaScript Applicare script a elementi DOM level Applicare script di validazione a moduli HTML Applicare il meccanismo degli eventi Applicare l'array elements[] 	<ul style="list-style-type: none"> Far interagire oggetti riflessi dal browser e script Saper progettare e codificare web application sfruttando l'interazione con i form Realizzare applicazioni con la validazione dell'input dell'utente all'interno dei form 	<ul style="list-style-type: none"> Realizzazione di progetti web con l'interazione tra i form e Javascript 	<ul style="list-style-type: none"> Inglese: linguaggio tecnico Esercizi che partano da argomenti affrontati nelle altre discipline 	maggio - giugno

Si propongono come eventuali approfondimenti a scelta dei docenti i seguenti argomenti presenti nel libro di testo in adozione:

- Unità 3: Sviluppo di applicazioni per dispositivi mobili;
- Unità 4: Strutture dati dinamiche;
- Unità 5: Archivi su memoria permanente;
- Unità 7: Principi teorici della computazione.



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

OBIETTIVI MINIMI DISCIPLINARI	Le conoscenze relative agli obiettivi minimi sono quelle riportate nel curriculum, ma in contesti basilari ed accettabili a livello di approfondimento/difficoltà e con competenze/abilità minime o parziali. Dagli obiettivi minimi si possono ulteriormente escludere i seguenti argomenti: il modulo 4 e gli eventuali approfondimenti scelti dai singoli docenti.	
APPROCCIO DIDATTICO COMUNE A TUTTI I MODULI	METODOLOGIE	STRUMENTI
	<ul style="list-style-type: none">• Didattica laboratoriale.• Lezione frontale.• Lezione dialogata.• Classe capovolta.• Apprendimento per scoperta.• Apprendimento per progetti.• Lavoro di gruppo• Lavoro individuale, interventi personalizzati.• Brainstorming.• Flipped classroom.	<ul style="list-style-type: none">• Compiti in classe• interrogazioni alla cattedra• interrogazioni al posto• interrogazioni programmate• questionari (prove oggettive)• relazioni da svolgere a casa• Lavagna.• Uso di PC in Laboratorio.• Piattaforma di e-learning.• Appunti elaborati dal docente.• Libro di testo.• Manuali tecnici.• Verifiche orali e scritte.• Test a risposta chiusa e aperta.• Sussidi audiovisivi.• Ambiente di sviluppo integrato (IDE).