



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

Curricolo verticale della disciplina

INFORMATICA

Percorso di studio: **Liceo scientifico - opzione scienze applicate**

SECONDO BIENNIO

TERZO ANNO

MODULO	PREREQUISITI	CONOSCENZE	ABILITÀ	COMPETENZE	ATTIVITÀ DI LABORATORIO	CONNESSIONI INTERDISCIPLINARI	TEMPI
MODULO 1 LA SCOMPOSIZIONE IN SOTTOPROGRAMMI	<ul style="list-style-type: none">• Programmare il C++	<ul style="list-style-type: none">• Comprendere il meccanismo del passaggio dei parametri• Comprendere le differenze tra il passaggio per indirizzo e per valore• Comprendere le regole di visibilità• Individuare un problema ricorsivo• Comprendere la differenza tra ricorsione e iterazione• Comprendere il concetto di ricorsione tail	<ul style="list-style-type: none">• Scrivere algoritmi usando le funzioni• Utilizzare funzioni predefinite nei programmi• Utilizzare funzioni personali• Scrivere funzioni ricorsive• Trasformare funzioni iterative in ricorsive	<ul style="list-style-type: none">• Definire una funzione• Definire la modalità del passaggio dei parametri• Distinguere i parametri formali e attuali• Organizzare un programma con menù	<ul style="list-style-type: none">• Attività di programmazione in C++	<ul style="list-style-type: none">• Matematica, Fisica e Chimica: scrivere in C++ problemi scientifici	Settembre - Novembre
MODULO 2 DATI STRUTTURATI E ALGORITMI CLASSICI	<ul style="list-style-type: none">•	<ul style="list-style-type: none">• Saper gestire dati di tipo diverso in relazione tra loro• Avere il concetto di archivio• Conoscere le tipologie di accesso ai dati• Riconoscere l'importanza dell'archiviazione dei dati• Conoscere la gestione dei dati sul supporto di massa	<ul style="list-style-type: none">• Scrivere, leggere e ricercare dati da un vettore• Scrivere e leggere dati da una matrice quadrata• Utilizzare array a due dimensioni• Elaborare le matrici con simmetrie• Manipolare record di tipi semplici• Operare con stringhe e sottostringhe	<ul style="list-style-type: none">• Saper organizzare tipi di dati complessi• Definire array monodimensionali e bidimensionali• Ricercare le simmetrie in una matrice quadrata• Definire una stringa• Definire record e strutture complesse di dati	<ul style="list-style-type: none">• Attività di programmazione in C++	<ul style="list-style-type: none">• Matematica, Fisica e Chimica: scrivere in C++ problemi scientifici	Novembre - Febbraio



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

MODULO 3 I FILE **	<ul style="list-style-type: none"> • 	<ul style="list-style-type: none"> • Avere il concetto di archivio • Conoscere tipologie di accesso ai dati • Riconoscere l'importanza dell'archiviazione dei dati • Conoscere la gestione dei dati sul supporto di massa • Conoscere tipologie di accesso ai dati 	<ul style="list-style-type: none"> • Scrivere, leggere e ricercare dati da un file • Effettuare l'accesso diretto ai dati • Creare archivi mediante file di record • Progettare applicazioni che utilizzano file 	<ul style="list-style-type: none"> • Distinguere file di testo da file binari • Saper organizzare tipi di dati complessi di memoria di massa • Definire un file sequenziale e binario • Definire e utilizzare file strutturati 	<ul style="list-style-type: none"> • 	<ul style="list-style-type: none"> • 	Febbraio
MODULO 4 LA PROGRAMMAZIONE AD OGGETTI E LE INTERFACCE GRAFICHE	<ul style="list-style-type: none"> • 	<ul style="list-style-type: none"> • Conoscere gli elementi teorici del paradigma a oggetti (OOP) • Comprendere il concetto di astrazione • Acquisire il concetto di costruzione e distruttore • Comprendere le differenze tra overloading e overriding • Conoscere una metodologie di documentazione delle classi (UML) • Conoscere il significato di classe astratta • Acquisire il concetto di programmazione a oggetti con le finestre grafiche • Conoscere i controlli: contenitori e componenti 	<ul style="list-style-type: none"> • Usare la progettazione orientata agli oggetti per programmi complessi • Applicare il concetto di astrazione per modellare le classi • Individuare la specializzazione e la generalizzazione di una classe • Applicare i concetti di ereditarietà e polimorfismo • Definire gerarchie di classi • Realizzar progetti con componenti grafici GUI • Utilizzare i controlli e gli oggetti 	<ul style="list-style-type: none"> • Definire una classe con attributi e metodi • Definire i costruttori e il distruttore di una classe • Classificare classi e relazioni tra di esse • Applicare i concetti di incapsulamento e information hiding • Riconoscere la gerarchia delle classi • Rappresentare classi e oggetti mediante diagrammi UML • Distinguere tra classi e oggetti, controlli ed eventi • Distinguere le tipologie di controllo adatte alle attività da svolgere 	<ul style="list-style-type: none"> • Attività di programmazione in C++ 	<ul style="list-style-type: none"> • Matematica, Fisica e Chimica: scrivere in C++ problemi scientifici 	Aprile – Maggio

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.	
APPROCCIO DIDATTICO COMUNE A TUTTI I MODULI	METODOLOGIE	STRUMENTI



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

	<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.• Flippedclassroom.	<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).
--	--	---

** Il modulo 3 è da considerarsi facoltativo