



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

## Curricolo verticale della disciplina **TECNOLOGIE E PROGETTAZIONE DI SISTEMI INFORMATICI E DI TELECOMUNICAZIONI**

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
<b>MODULO 1</b> Fasi e modelli di gestione di un ciclo di sviluppo	•	<ul style="list-style-type: none"><li>• Comprendere la necessità di una metodologia per lo sviluppo di sistemi informatici</li><li>• Conoscere gli elementi fondamentali dell'ingegneria del software</li><li>• Conoscere gli elementi teorici della progettazione a oggetti (OOP)</li><li>• Capire l'utilizzo delle schede CRC per l'identificazione di classi</li><li>• Conoscere una metodologia di documentazione (UML)</li><li>• Conoscere le tecniche di schedulazione delle fasi</li></ul>	<ul style="list-style-type: none"><li>• Scegliere le metodologie e le tecniche adeguate alle diverse situazioni</li><li>• Applicare il concetto di astrazione per modellare le classi</li><li>• Utilizzare ArgoUML per documentare un progetto</li><li>• Utilizzare Excel per realizzare diagrammi di Gantt</li></ul>	<ul style="list-style-type: none"><li>• Individuare e descrivere il problema complesso</li><li>• Usare la progettazione orientata agli oggetti per programmi complessi</li><li>• Rappresentare classi e oggetti mediante diagrammi UML</li><li>• Usare i diagrammi UML per descrivere le relazioni tra gli elementi di un progetto</li><li>• Usare la progettazione orientata agli oggetti per sistemi informatici complessi</li><li>• Schedulare le fasi di un progetto con i diagrammi di Gantt</li></ul>	<ul style="list-style-type: none"><li>• Realizzare un diagramma di Gantt con Excel</li><li>• I diagrammi UML con software</li></ul>		Settembre-Ottobre



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

<p><b>MODULO 2</b> <b>I REQUISITI SOFTWARE</b></p>		<ul style="list-style-type: none"> <li>• Comprendere l'importanza della fase di analisi</li> <li>• Avere il concetto di requisito utente e di sistema</li> <li>• Avere il concetto di fase di esplorazione</li> <li>• Conoscere le tecniche di esplorazione</li> <li>• Avere il concetto di scenario e caso d'uso</li> <li>• Comprendere i principi S.O.L.I.D. per la definizione delle classi</li> </ul>	<ul style="list-style-type: none"> <li>• Saper descrivere in UML i casi d'uso</li> <li>• Saper descrivere in UML il diagramma di contesto</li> <li>• Saper documentare i casi d'uso</li> </ul>	<ul style="list-style-type: none"> <li>• Individuare i requisiti utente e di sistema</li> <li>• Utilizzare le tecniche di esplorazione</li> <li>• Individuare gli scenari d'uso</li> </ul>	<ul style="list-style-type: none"> <li>• Uso dei diagrammi di stato , attività (progettazione algoritmo) in UML e tramite Software applicativo.</li> </ul>	<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Novembre-Dicembre</p>
<p><b>MODULO 3</b> <b>DOCUMENTAZIONE DEL SOFTWARE</b></p>		<ul style="list-style-type: none"> <li>• Comprendere la necessità di documentare</li> <li>• Sapere quali sono i documenti necessari in un progetto</li> <li>• Conoscere il concetto di documentazione interna ed esterna</li> <li>• Apprendere le modalità per realizzarla documentazione esterna di sistema e utente</li> <li>• Acquisire una tecnica di documentazione del codice</li> <li>• Conoscere i principali tool di documentazione automatica del codice</li> </ul>	<ul style="list-style-type: none"> <li>• Saper usare i principali tool di documentazione automatica del codice</li> <li>• Saper definire uno standard di documentazione</li> </ul>	<ul style="list-style-type: none"> <li>• Saper organizzare la documentazione del progetto</li> <li>• Saper formattare il codice</li> <li>• Saper effettuare la documentazione del codice</li> <li>• Saper condividere componenti di un progetto con il team di sviluppo</li> </ul>	<ul style="list-style-type: none"> <li>• Utilizzare Javadoc come strumento di documentazione automatica</li> </ul>	<ul style="list-style-type: none"> <li>• Informatica: Conoscenza della Classi e loro principale Uso. Analisi, studio dei dati principali e stesura di algoritmi e UML</li> </ul>	<p>Gennaio-Febbraio</p>
<p><b>MODULO 4</b> <b>PROCESSI SEQUENZIALI E PARALLELI</b></p>		<ul style="list-style-type: none"> <li>• Conoscere i modelli di elaborazione dei processi</li> <li>• Conoscere il ciclo di vita dei processi</li> <li>• Acquisire il concetto di risorsa condivisa</li> <li>• Distinguere le richieste e le modalità di accesso alle risorse</li> <li>• Apprendere l'utilizzo del grafo di Holt per descrivere i processi</li> <li>• Conoscere la differenza tra processo e thread</li> <li>• Acquisire il concetto di programmazione concorrente</li> <li>• Acquisire il concetto di interazione tra processi</li> <li>• Conoscere le caratteristiche di un linguaggio concorrente</li> </ul>	<ul style="list-style-type: none"> <li>• Utilizzare i thread in Java</li> <li>• Conoscere i metodi sleep(), yield() e join() in Java</li> </ul>	<ul style="list-style-type: none"> <li>• Descrivere l'interazione processi-risorse con il grafo di Holt</li> <li>• Realizzare e semplificare il grafo delle precedenze</li> <li>• Scrivere in pseudocodifica programmi concorrenti utilizzando l'istruzione fork-join</li> <li>• Scrivere in pseudo codifica programmi concorrenti utilizzando l'istruzione cobegin-coend</li> </ul>	<ul style="list-style-type: none"> <li>• Scrivere programmi concorrenti in Java</li> </ul>	<ul style="list-style-type: none"> <li>• Informatica: linguaggio Java</li> </ul>	<p>Febbraio-Marzo</p>



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

<p><b>MODULO 5 COMUNICAZIONE E SINCRONIZZAZIONE</b></p>		<ul style="list-style-type: none"> <li>• Conoscere il modello ad ambiente globale e locale</li> <li>• Comprendere l'esigenza di sincronizzazione</li> <li>• Comprendere il concetto di indivisibilità di una primitiva</li> <li>• Sapere il funzionamento dei semafori di Dijkstra</li> <li>• Avere il concetto di regione critica e mutua esclusione</li> <li>• Sapere la differenza tra interleaving e overleaving</li> <li>• Comprendere le condizioni di Bernstein</li> <li>• Avere il concetto di starvation e di deadlock</li> <li>• Comprendere le proprietà di safety, di fairness e di liveness</li> </ul>	<ul style="list-style-type: none"> <li>• Risolvere le situazioni di starvation</li> <li>• Risolvere le situazioni di deadlock</li> </ul>	<ul style="list-style-type: none"> <li>• Individuare le tipologie di errori nei processi paralleli</li> <li>• Definire e utilizzare i semafori di basso livello e spinlock()</li> <li>• Implementare i monitor di Hoare in Java</li> </ul>	<ul style="list-style-type: none"> <li>• I semafori e i monitor in Java</li> <li>• I deadlock in Java</li> </ul>		<p>Aprile – Maggio</p>
<p><b>MODULO 6**</b> Arduino o Raspberry Pi Interazione tra HW e SW</p>		<ul style="list-style-type: none"> <li>• Progettare e costruire mini circuiti di base per l'interfacciamento tra la scheda Arduino ed il PC / o scheda RaspberryPi</li> </ul>	<ul style="list-style-type: none"> <li>• Simulazione di circuiti elettrici programmabili.</li> <li>• Implementazione del codice su IDE Arduino</li> </ul>	<ul style="list-style-type: none"> <li>• Arduino:</li> <li>• L'IDLE Arduino</li> <li>• Lo sketch blink.</li> <li>• Input digitali e comunicazioni seriali.</li> <li>• Input e Output analogici.</li> <li>• Prova di realizzazione di un progetto completo.</li> </ul>	<ul style="list-style-type: none"> <li>• Uso della scheda Arduino per provare i comandi principali con integrazione dei comandi del SO / o scheda RaspberryPi</li> </ul>	<ul style="list-style-type: none"> <li>• Integrazione con la materia Elettronica digitale.</li> </ul>	<p>Maggio – Giugno</p>

<p><b>OBIETTIVI MINIMI DISCIPLINARI</b></p>	<p>Le conoscenze relative agli obiettivi minimi sono quelle riportate nel curricolo, ma in contesti basilari ed accettabili a livello di approfondimento/difficoltà e con competenze/abilità minime o parziali.</p>	
<p><b>APPROCCIO DIDATTICO COMUNE A TUTTI I MODULI</b></p>	<p><b>METODOLOGIE</b></p>	<p><b>STRUMENTI</b></p>



Istituto di Istruzione Superiore "Giorgi – Woolf"

Viale Palmiro Togliatti, 1161 – 00155 Roma

	<ul style="list-style-type: none"><li>• Didattica laboratoriale.</li><li>• Lezione frontale.</li><li>• Lezione dialogata.</li><li>• Classe capovolta.</li><li>• Apprendimento per scoperta.</li><li>• Apprendimento per progetti.</li><li>• Lavoro di gruppo</li><li>• Lavoro individuale, interventi personalizzati.</li><li>• Brainstorming.</li><li>• Flipped classroom.</li></ul>	<ul style="list-style-type: none"><li>• Compiti in classe</li><li>• interrogazioni alla cattedra</li><li>• interrogazioni al posto</li><li>• interrogazioni programmate</li><li>• questionari (prove oggettive)</li><li>• relazioni da svolgere a casa</li><li>• Lavagna.</li><li>• Uso di PC in Laboratorio.</li><li>• Piattaforma di e-learning.</li><li>• Appunti elaborati dal docente.</li><li>• Libro di testo.</li><li>• Manuali tecnici.</li><li>• Verifiche orali e scritte.</li><li>• Test a risposta chiusa e aperta.</li><li>• Sussidi audiovisivi.</li><li>• Ambiente di sviluppo integrato (IDE).</li></ul>
--	---	---