



**iis**  
**MarzottoLuzzatti**  
**Valdagno**

Sito Internet: [www.iisvaldagno.it](http://www.iisvaldagno.it)

E-Mail: [viis022004@istruzione.it](mailto:viis022004@istruzione.it)

E-Mail Certificata: [viis022004@pec.istruzione.it](mailto:viis022004@pec.istruzione.it)



## PROGRAMMA EFFETTIVAMENTE SVOLTO

Anno Scolastico 2021 / 2022

**Classe/Sede:** 4D2 sede ITI

**Docente:** Pugliese Vincenzo

**Codocente (ITP):** Fasulo Simone Manlio

**Materia insegnata:** Tecnologie Progettazione Sistemi Informatici e Telecomunicazioni (TPSIT)

**Testi adottati:** Tecnologie Progettazione Sistemi Informatici e di Telecomunicazioni ,*seconda edizione* , Ciclo di vita del software e UML, Programmazione Concorrente

Autori: Giorgio Meini, Fiorenzo Formichi

Editore: Zanichelli

### CONTENUTI DISCIPLINARI

#### **Modulo 1 :Ciclo di vita del software, UML, gestione e documentazione del codice**

1.1 – Ciclo di vita e ingegneria del software: metodologie di sviluppo; linguaggio di modellizzazione UML; qualità del sw e pattern( es less is more, keep it simple, Make it fast, make it run etc)

1.2 – Requisiti software e casi d'uso: definizione e classificazione dei requisiti di un prodotto/servizio software; diagrammi UML dei casi d'uso, sequence diagram, activity diagram, class diagram e loro rappresentazione in Visio o altri software. Progettazione del software: esercizi sui casi d'uso assegnati e altri esempi su V-Learning.

1.3 – Gestione e documentazione del software, commenti interni, regole e convenzioni di codifica C++, Google, Linux e standard MISRA ; ambienti di sviluppo integrati IDE, diritto d'autore e licenze software; documentazione del codice sorgente con Doxygen; gestione delle versioni del codice sorgente, introduzione a Github e problemi della gestione delle versioni e modifiche su repository remote.

Bug software: casi significativi di bug e disastri storici. Ricerca dei bug, test di vario tipo, debugging.

1.5 Test del software: pianificazione e classificazione dei test; unit test , Casi di test. Test alfa, beta, White e black box, di regressione, di integrazione, di prestazione, di load e stress. Sistemi critici, esempi di essi , test di sistemi critici. Sistemi di tracciamento dei bug: esempio di BugZilla. Parole chiave ASSERT ed EXPECT per casi di test UNITARI.

**Attività pratiche in classe e in laboratorio:** esercitazioni su diagrammi UML utilizzando varie applicazioni (MS Visio, Lucid Charts o Microsoft Draw (disegni tecnici gratuiti); documentare i propri progetti software (sviluppati in questa o in altre materie) tramite Doxygen; utilizzare i comandi di Git; creare un account e

utilizzare GitHub, caricando i propri progetti. Gestire una o più branch indipendenti. Effettuare i commit, check out, e eventuale merge.

## Modulo 2:

### Linguaggio C/C#: ripasso e uso nella programmazione concorrente IPC

1. 2.1 – Ripasso: linguaggio C; esercizi vari di ripasso con i costrutti principali, printf e scanf, metodo main(), dichiarazione di tipi atomici, allocazione dati, operazioni elementi es addizioni, sottrazione, metodi in c, prototipi, strutture dati, puntatori, etc. I file e loro gestione: lettura/scrittura di dati su/da file, file di testo e file binari, metodi a confronto.

2.2 – Inter-Process Communication in ambiente Linux, I processi in Linux; differenza tra processo e thread, librerie da includere, primitive principali fork, wait, pipe; esecuzione di programmi e FIFO; memoria condivisa e semafori; schema produttore-consumatore; memoria condivisa tra processi indipendenti; schema lettori-scrittori; cenni a problematiche di deadlock.

2.3 – Programmazione multi-thread: Thread e tipi atomici, mutex e loro uso, introduzione ai thread e alla programmazione concorrente e multiprocessore, problematiche di sincronizzazione, esempi di thread in C#  
*Riferimenti libro di testo:* capitoli B1 e B3.

Valdagno, 06/06/2022

*Firma degli studenti  
rappresentanti di classe*

\_\_\_\_\_  
\_\_\_\_\_

Firma dei Docenti

*Vincenzo Pugliese*

*Fasulo Simone Manlio*