

PROGRAMMAZIONE TPS CLASSE 4°

Disciplina: TECNOLOGIE E PROGETTAZIONE DI SISTEMI INFORMATICI E DI TELECOMUNICAZIONI

Classe: **4** I.T.I.S corso Informatica Anno Scolastico 2016/15

Finalità:

- Ideazione e realizzazione di progetti informatici;
- scegliere dispositivi e strumenti in base alle loro caratteristiche funzionali;
- gestire progetti utilizzando gli strumenti di progettazione, documentazione e controllo;

Obiettivi:

Gli obiettivi della programmazione curricolare annuale si propongono di fornire all'allievo le basi teoriche e pratiche per:

- realizzare progetti di sistemi informatici,
- applicare le tecnologie informatiche per la programmazione concorrente,
- analizzare e progettare applicazioni parallele, con lo studio delle problematiche di sincronizzazione e soluzione dei casi di deadlock e starvation,
- definire tecniche e metodologie di collaudo e documentazione del progetto di sistemi informatici.

Contenuti:

Ciclo di vita del software e UML:

- Ingegneria del software metodologie di sviluppo
- Linguaggio di modellizzazione UML

Documentazione e test del software:

- Regole e convenzioni di codifica
- ambienti di sviluppo integrati
- pianificazione e classificazione dei test
- strumenti per la codifica e l'esecuzione di test unitari

Programmazione Concorrente:

- Thread in Java
- Condivisione di risorse fra thread
- Sincronizzazione dei thread
- Cooperazione e competizione fra processi
- I semafori
- Il modello produttore consumatore

Arduino:

- L'IDLE Arduino
- Lo sketch blink
- Input digitali e comunicazioni seriali
- Input e Output analogici
- Un progetto completo

Modalità di lavoro:

1. Spiegazioni in classe e/o in laboratorio.
2. Utilizzo del laboratorio di informatica e del laboratorio di assemblaggio.
3. Studio e svolgimento di esercizi a casa.
4. Assegnazione di compiti con stesura di relazioni.
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 e laboratorio di assemblaggio.
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) ogni studente sarà sottoposto mediamente a 2 prove di verifica sommativa di tipologia varia.

Secondo periodo (pentamestre) ogni studente sarà sottoposto mediamente a 4 prove di verifica sommativa di tipologia varia.

Requisiti minimi per l'attribuzione della sufficienza:

Lo studente:

- possiede i contenuti minimi: basi delle metodologie di sviluppo, modellizzazione e test del software, della programmazione concorrente, della programmazione della scheda Arduino.
- 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.

N°	Modulo	Obiettivi	Tempi	Metodologie	Valutazione
1	Ciclo di vita del software e UML	Definire metodologie di documentazione del software	20 h	lezioni frontali lavoro di gruppo	prove strutturate interrogazioni
2	Documentazione e test del software	Definire metodologie di test del software	10 h	lezioni frontali lavoro di gruppo attività di lab	prove strutturate interrogazioni
3	Programmazione Concorrente	progettare applicazioni parallele, con lo studio delle problematiche di sincronizzazione	40 h	lezioni frontali lavoro di gruppo attività di lab	prove strutturate interrogazioni
4	Arduino	realizzare progetti di sistemi informatici	30h	lezioni frontali lavoro di gruppo attività di lab	interrogazioni prove strutturate, progetti

DEFINIZIONE DEI REQUISITI MINIMI PER L'ATTRIBUZIONE DELLA SUFFICIENZA

MATERIA: TECNOLOGIE E PROGETTAZIONE DI SISTEMI INFORMATICI E DI

TELECOMUNICAZIONI CLASSE: **3**

CONTENUTI	CONOSCENZE MINIME	COMPETENZE MINIME
• Concetto di informazione • Basi dei sistemi di numerazione • Connettivi logici fondamentali • Le principali codifiche	• Conoscere il concetto di informazione • Conoscere le trasformazioni da una base di numerazione ad un'altra • Conoscere i calcoli in complemento a due • Conoscere i calcoli in floating point • Conoscere le principali codifiche	• 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
• Hardware di un pc	Conoscere le basi dell'architettura di un elaboratore	
• Elementi base dei sistemi operativi	• Conoscere le funzioni del sistema operativo • Conoscere i principali moduli del sistema operativo	
• linguaggio C	• Conoscere le basi del linguaggio C	