|  |
| --- |
| ***Premessa:*** |
| La disciplina “Tecnologie e progettazione di sistemi informatici e di telecomunicazioni ” concorre a far conseguire allo studente al termine del percorso quinquennale i seguenti risultati di apprendimento relativi al profilo educativo, culturale e professionale **(PECUP)**:   * Orientarsi nella normativa che disciplina i processi produttivi del settore di riferimento, con particolare attenzione sia alla sicurezza sui luoghi di vita e di lavoro sia alla tutela dell’ambiente e del territorio. * Intervenire nelle diverse fasi e livelli del processo produttivo, dall’ideazione alla realizzazione del prodotto, per la parte di propria competenza, utilizzando gli strumenti di progettazione, documentazione e controllo. * Riconoscere gli aspetti di efficacia, efficienza e qualità nella propria attività lavorativa. * Padroneggiare l’uso di strumenti tecnologici con particolare attenzione alla sicurezza nei luoghi di vita e di lavoro, alla tutela della persona, dell’ambiente e del territorio.   **Competenze Disciplinari**   * Sviluppare applicazioni informatiche per reti locali o servizi a distanza. * Scegliere dispositivi e strumenti in base alle loro caratteristiche funzionali. * Gestire progetti secondo le procedure e gli standard previsti dai sistemi aziendali di gestione della qualità e della sicurezza, gestire processi produttivi correlati a funzioni aziendali. * Configurare, installare e gestire sistemi di elaborazione dati e reti. * Redigere relazioni tecniche e documentare le attività individuali e di gruppo relative a situazioni professionali.   La metodologia utilizzata sarà strettamente connessa all'individuazione degli obiettivi e dei contenuti, di particolare importanza è la scelta delle strategie operative e dei mezzi più idonei alla realizzazione del processo di apprendimento da parte degli alunni. In tale ottica, gli strumenti della mediazione didattica terranno conto del contesto e assicureranno la sostenibilità delle attività proposte e un generale livello di inclusività.  Per quanto non espressamente esplicitato si fa riferimento ai Verbali di dipartimento ratificati dal Collegio dei Docenti per il corrente anno scolastico (standard di valutazione, verifica trasversale per “Assi culturali” e classi parallele, attività interdisciplinari, visite, viaggi, sopralluoghi aziendali, attività formative extracurricolari, P.T.O.F.) |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modulo1: TEORIA E CODIFICA DELL’INFORMAZIONE** | | | | **Periodo: Settembre - Dicembre** | | |
| **Competenze disciplinari** | **Livelli di apprendimento** | **Conoscenze** | **Abilità** | | **Indicazioni**  **Metodologiche** | **Attività e Verifiche** |
| Convertire numeri e codici rappresentati secondo sistemi diversi.  Convertire un numero in base decimale, binario, ottale ed esadecimale.  Distinguere le modalità di codifica dei suoni  Conoscere il codice ASCII e Unicode.  Codificare in codice BCD, Eccesso 3 e Gray.  Codificare a sette segmenti e a matrice di punti.  Saper codificare con i codici di Hamming e rilevare eventuali errori di trasmissione  Codificare i numeri in modulo e segno.  Codificare e decodificare i numeri in IEEE-P75. | **A - Livello di apprendimento avanzato** | Acquisire il concetto di comunicazione.  Conoscere il concetto di alfabeto, codifica e protocollo.  Comprendere la differenza tra segnale analogico e digitale.  Sistema di numerazione decimale, binario, ottale, esadecimale.  Codifica di immagini, suoni e filmati  Comprendere le differenze tra codifica a lunghezza fissa e variabile.  Conoscere le codifiche per dispositivi dedicati.  Comprendere le tecniche di rilevazione e di correzione degli errori di trasmissione  Acquisire la nozione di complemento di un numero.  Acquisire il concetto di overflow.  Conoscere lo standard IEEE-P754 a 32 e 64 bit. | Effettuare la conversione da decimale a basi pesate di numeri interi e frazionali.  Calcolare l’occupazione di memoria di immagini e suoni digitali  Eseguire somma e sottrazione in BCD.  Correggere l’errore con byte di checksum.  Individuare l’errore con il codice di parità.  Correggere l’errore con il codice di Hamming  Eseguire il complemento a 1 e a 2 di un numero binario.  Effettuare le operazioni algebriche tra numeri binari.  Rappresentare i numeri decimali in virgola mobile. | | Lezione frontale soprattutto nei moduli a carattere portante. lezione interattiva. Problem solving con connotazioni di ricerca-azione nello studio di problemi. | Interrogazione individuale.  Verifica formativa.  Test online.  Prove pratiche. |
| **B - Livello di apprendimento intermedio** | Acquisire il concetto di comunicazione.  Conoscere il concetto di alfabeto, codifica e protocollo.  Comprendere la differenza tra segnale analogico e digitale.  Sistema di numerazione decimale, binario, ottale, esadecimale.  Comprendere le differenze tra codifica a lunghezza fissa e variabile.  Conoscere le codifiche per dispositivi dedicati.  Comprendere le tecniche di rilevazione e di correzione degli errori di trasmissione  Acquisire la nozione di complemento di un numero.  Acquisire il concetto di overflow. | Effettuare la conversione da decimale a basi pesate di numeri interi e frazionali.  Calcolare l’occupazione di memoria di immagini e suoni digitali  Eseguire somma e sottrazione in BCD.  Correggere l’errore con byte di checksum.  Individuare l’errore con il codice di parità.  Correggere l’errore con il codice di Hamming  Eseguire il complemento a 1 e a 2 di un numero binario. | |
| **C - Livello di apprendimento base** | Acquisire il concetto di comunicazione.  Conoscere il concetto di alfabeto, codifica e protocollo.  Comprendere la differenza tra segnale analogico e digitale.  Sistema di numerazione decimale, binario, ottale, esadecimale.  Comprendere le differenze tra codifica a lunghezza fissa e variabile.  Acquisire la nozione di complemento di un numero.  Acquisire il concetto di overflow. | Effettuare la conversione da decimale a basi pesate di numeri interi e frazionali.  Calcolare l’occupazione di memoria di immagini e suoni digitali  Correggere l’errore con byte di checksum.  Individuare l’errore con il codice di parità.  Eseguire il complemento a 1 e a 2 di un numero binario. | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modulo 2: MODELLI E AUTOMI** | | | | **Periodo: Dicembre – Marzo** | | |
| **Competenze disciplinari** | **Livelli di apprendimento** | **Conoscenze** | **Abilità** | | **Indicazioni**  **Metodologiche** | **Attività e Verifiche** |
| Riconoscere il ruolo dei sistemi e degli automi. Definire e comprendere funzioni di transizione e di trasformazione.  Identificare la struttura di automi di Mealy e di Moore.  Realizzare graficamente automi di Mealy e di Moore.  Utilizzare la scheda Arduino per la realizzazione di alcuni automi. | **A - Livello di apprendimento avanzato** | Conoscere la definizione formale di sistema. Conoscere la definizione di modello.  Conoscere i concetti di parametro e variabile, stato interno e memoria di un sistema, ingresso, uscita, funzione di transizione e di trasformazione.  Conoscere la definizione di automa.  Conoscere la scheda Arduino ed i sensori ad essa collegabili. | Saper classificare sistemi e modelli.  Saper rappresentare un sistema tramite un modello.  Saper distinguere le grandezze continue da quelle discrete.  Saper comporre le tabelle di transizione e trasformazione.  Saper rappresentare un automa a stati finiti.  Saper realizzare un automa riconoscitore di sequenze  Saper realizzare programmi in linguaggio C per la programmazione della scheda Arduino. | | Lezione frontale soprattutto nei moduli a carattere portante. lezione interattiva. problem solving con connotazioni di ricerca-azione nello studio di problemi. | Interrogazione individuale.  Verifica formativa.  Test online.  Prove pratiche. |
| **B - Livello di apprendimento intermedio** | Conoscere la definizione formale di sistema. Conoscere la definizione di modello.  Conoscere i concetti di parametro e variabile, stato interno e memoria di un sistema, ingresso, uscita, funzione di transizione e di trasformazione.  Conoscere la definizione di automa.  Saper progettare semplici automi. | Saper classificare sistemi e modelli.  Saper rappresentare un sistema tramite un modello.  Saper distinguere le grandezze continue da quelle discrete.  Saper comporre le tabelle di transizione e trasformazione.  Saper rappresentare un automa a stati finiti.  Saper realizzare un automa riconoscitore di sequenze. | |
| **C - Livello di apprendimento base** | Conoscere la definizione formale di sistema. e di modello.  Conoscere i concetti di parametro e variabile, stato interno e memoria di un sistema, ingresso, uscita, funzione di transizione e di trasformazione.  Conoscere la definizione di automa. | Saper classificare sistemi e modelli.  Saper rappresentare un sistema tramite un modello.  Saper distinguere le grandezze continue da quelle discrete.  Saper comporre le tabelle di transizione e trasformazione. | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modulo 3: TEORIA DEI SISTEMI OPERATIVI** | | | | **Periodo: Marzo - Maggio** | | |
| **Competenze disciplinari** | **Livelli di apprendimento** | **Conoscenze** | **Abilità** | | **Indicazioni**  **Metodologiche** | **Attività e Verifiche** |
| Classificare i sistemi operativi.  Descrivere il ciclo di vita di un processo.  Classificare le memorie.  Riconoscere il modello client-server.  Classificare le tecniche di gestione delle periferiche. | **A - Livello di apprendimento avanzato** | Sapere che cosa succede all’accensione del PC.  Conoscere i compiti del sistema operativo.  Conoscere la storia dei sistemi operativi.  Riconoscere i meccanismi di caricamento del programma in memoria.  Conoscere le tecniche di virtualizzazione della memoria.  Descrivere le tecniche di realizzazione del file system.  I sistemi di protezione dei dati.  Conoscere l’hardware dei dispositivi di I/O. | Utilizzare in modo appropriato la terminologia tecnica.  Riconoscere le caratteristiche principali del sistema operativo.  Scegliere le politiche di allocazione del processore.  Individuare le problematiche per la cooperazione tra processi.  Utilizzare le tecniche di back-up dei dati. | | Lezione frontale soprattutto nei moduli a carattere portante. lezione interattiva. problem solving con connotazioni di ricerca-azione nello studio di problemi. | Interrogazione individuale.  Verifica formativa.  Test online.  Prove pratiche. |
| **B - Livello di apprendimento intermedio** | Sapere che cosa succede all’accensione del PC.  Conoscere i compiti del sistema operativo.  Conoscere la storia dei sistemi operativi.  Riconoscere i meccanismi di caricamento del programma in memoria.  Conoscere le tecniche di virtualizzazione della memoria.  Descrivere le tecniche di realizzazione del file system. | Utilizzare in modo appropriato la terminologia tecnica.  Riconoscere le caratteristiche principali del sistema operativo.  Scegliere le politiche di allocazione del processore.  Utilizzare le tecniche di back-up dei dati. | |
| **C - Livello di apprendimento base** | Sapere che cosa succede all’accensione del PC.  Conoscere i compiti del sistema operativo.  Conoscere la storia dei sistemi operativi.  Riconoscere i meccanismi di caricamento del programma in memoria. | Utilizzare in modo appropriato la terminologia tecnica.  Riconoscere le caratteristiche principali del sistema operativo.  Utilizzare le tecniche di back-up dei dati. | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modulo 4: FASI E MODELLI DI GESTIONE DI UN CICLO DI SVILUPPO** | | | | **Periodo: Maggio – Giugno** | | |
| **Competenze disciplinari** | **Livelli di apprendimento** | **Conoscenze** | **Abilità** | | **Indicazioni**  **Metodologiche** | **Attività e Verifiche** |
| Conoscere gli elementi fondamentali dell’ingegneria del software.  Rappresentare classi e oggetti mediante diagrammi UML.  Usare i diagrammi UML per descrivere le relazioni tra gli elementi di un progetto.  Schedulare le fasi di un progetto con i diagrammi di Gantt. | **A - Livello di apprendimento avanzato** | Comprendere le necessità di una metodologia per lo sviluppo di sistemi informatici.  Conoscere gli elementi fondamentali dell’ingegneria del software.  Conoscere gli elementi teorici della progettazione a oggetti (OOP).  Conoscere una metodologia di documentazione (UML).  Conoscere le tecniche di schedulazione delle fasi. | Scegliere le metodologie e le tecniche adeguate alle diverse situazioni.  Applicare il concetto di astrazione per modellare le classi.  Utilizzare le principali funzioni ArgoUML.  Utilizzare uno specifico software per realizzare i diagrammi di Gantt | | Lezione frontale soprattutto nei moduli a carattere portante. lezione interattiva. problem solving con connotazioni di ricerca-azione nello studio di problemi. | Interrogazione individuale.  Verifica formativa.  Test online.  Prove pratiche. |
| **B - Livello di apprendimento intermedio** | Comprendere le necessità di una metodologia per lo sviluppo di sistemi informatici.  Conoscere gli elementi fondamentali dell’ingegneria del software.  Conoscere gli elementi teorici della progettazione a oggetti (OOP).  Conoscere le tecniche di schedulazione delle fasi. | Scegliere le metodologie e le tecniche adeguate alle diverse situazioni.  Applicare il concetto di astrazione per modellare le classi.  Sapere i principali software per rappresentare l’UML.  Utilizzare uno specifico software per realizzare i diagrammi di Gantt. | |
| **C - Livello di apprendimento base** | Comprendere le necessità di una metodologia per lo sviluppo di sistemi informatici.  Conoscere gli elementi fondamentali dell’ingegneria del software.  Conoscere gli elementi teorici della progettazione a oggetti (OOP). | Scegliere le metodologie e le tecniche adeguate alle diverse situazioni.  Applicare il concetto di astrazione per modellare le classi.  Sapere i principali software per realizzare i diagrammi di Gantt. | |