| **PROGRAMMA DIDATTICO CLASSE IIIC MATERIA INFORMATICA INDUSTRIALE** | | | |
| --- | --- | --- | --- |
| **Modulo 0** : Introduzione al problem solving | | | |
| **Prerequisiti:** Nessuno | | | |
| **Obiettivo generale dell’unità**: Introdurre gli strumenti formali per affrontare un problema e risolverlo | | | |
| 1. **Obiettivi** | 1. **Contenuti** | 1. **Metodi** | 1. **Periodo** |
| Introdurre lentamente lo studente alle modalità di analisi e risoluzione dei problemi  L’algoritmo  L’analisi  La soluzione | * L’analisi di un problema * Analisi del testo * Approccio sistematico, consigli * Scomposizione e semplificazione. * Caratteristiche di un algoritmo * Definizione * Conoscenza dei linguaggio di programmazione e tipologia * Analisi del problema * Comprensione del problema * Astrazione di un problema * Modelli grafici, tabellari e simbolici * Strumenti per la risoluzione * Strumenti algebrico, grafico, intuito * Sommatorie | * Lezione frontale * Lavoro in laboratorio su piccoli problemi da risolvere con metodo algoritmico * Lezione frontale * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Consultazione di documenti elettronici tratti da internet | Settembre  Ottobre |

| **PROGRAMMA DIDATTICO CLASSE IIIC MATERIA INFORMATICA INDUSTRIALE** | | | |
| --- | --- | --- | --- |
| **Modulo 1** : Introduzione all’approccio algoritmico dei problemi | | | |
| **Prerequisiti:** Nessuno | | | |
| **Obiettivo generale dell’unità**: Introdurre gli strumenti formali e le tecniche per costruire un algoritmo in base ad un problema dato | | | |
| 1. **Obiettivi** | 1. **Contenuti** | 1. **Metodi** | 1. **Periodo** |
| Introdurre lentamente lo studente alle modalità di trattamento dei problemi allo scopo di rendere possibile la risoluzione tramite algoritmi.  Studio delle strutture fondamentali  Spiegare l’iter dal problema al programma passando dall’algoritmo | * Approccio al problema con metodo sistematico e con la forza bruta * Tecniche di approccio utilizzando una modellazione algebrica, con la scomposizione, tramite l’analisi del testo * Concetto di top down e bottom up, tecniche per ridurre la complessità dei problemi. * Sequenza, iterazione, selezione * Formalizzazione di un algoritmo: pseudolinguaggio e diagramma di flusso * Corrispondenza fra diagramma di flusso e pseudolinguaggio, studio delle varianti. * Il concetto di programma * Sintassi, semantica, linguaggi * Compilazione, linking, esecuzione * L’ambiente di sviluppo per la programmazione | * Lezione frontale * Lavoro in laboratorio su piccoli problemi da risolvere con metodo algoritmico * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Consultazione di documenti elettronici tratti da internet | Settembre  Ottobre |

| **PROGRAMMA DIDATTICO CLASSE IIIC MATERIA INFORMATICA INDUSTRIALE** | | | |
| --- | --- | --- | --- |
| **Modulo 2** : Il linguaggio di programmazione e la programmazione imperativa | | | |
| **Prerequisiti:** Nessuno | | | |
| **Obiettivo generale dell’unità**: Permettere all’allievo la conoscenza delle informazioni di base per poter sviluppare algoritmi in un linguaggio di programmazione. | | | |
| 1. **Obiettivi** | 1. **Contenuti** | 1. **Metodi** | 1. **Periodo** |
| Consentire il passaggio dallo pseudo-linguaggio al codice  Migliorare la tecnica di programmazione in vista di programmi voluminosi | * Tipi semplici del linguaggio * Sintassi di base di un programma C * Concetto di istruzione, di operazione, di espressione * Traduzione di ogni struttura fondamentale studiata nel linguaggio, relazioni. * Ciclo while, ciclo for, selezione if, funzione main * Operatori && , ||, costrutto switch * Funzioni di input e output * Criteri per il nome delle variabili, incolonnamento del programma. * Un dato strutturato : l’array: indice e contenuto, programmi di ricerca, cancellazione, ordinamento * Strutture, array di strutture. * Typedef e strutture annidate * Stringhe, caratteristiche , funzioni tipiche di manipolazione * Array come puntatori, gli operatori \* e & * Matrici | * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Manuale del linguaggio * Lavoro singolo su carta * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Consultazione manuale * Lavoro singolo su carta | Novembre  Dicembre  Gennaio |

| **PROGRAMMA DIDATTICO CLASSE IIIC MATERIA INFORMATICA INDUSTRIALE** | | | |
| --- | --- | --- | --- |
| **Modulo 3** : I sottoprogrammi | | | |
| **Prerequisiti:** I moduli precedenti | | | |
| **Obiettivo generale dell’unità**: Far comprendere come l’aumento delle dimensioni di un programma richieda l’uso di sottoprogrammi strutturati in funzioni, specificare la costruzione e l’uso delle stesse nel linguaggio C | | | |
| 1. **Obiettivi** | 1. **Contenuti** | 1. **Metodi** | 1. **Periodo** |
| Le funzioni in C  Il passaggio dei parametri  Tecniche di programmazione per moduli | * Definizione di una funzione * Prototipi, header file * Tipi di ritorno * Passaggio parametri * Passaggio parametri, differenza fra passaggio per valore e per riferimento * Tipo puntatore, notazione e definizione * Passaggio di un parametro tramite l’indirizzo, uso e utilità * Incapsulamento * Vantaggi e svantaggi della programmazione per moduli * Criteri per la scelta della segnatura di una funzione | * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Libro di testo * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Consultazione di documenti elettronici tratti da internet * Libro di testo * Lezione frontale * Lavoro in laboratorio su piccoli programmi * Consultazione manuale del linguaggio | Marzo  Aprile |

| **PROGRAMMA DIDATTICO CLASSE IIIC MATERIA INFORMATICA INDUSTRIALE** | | | |
| --- | --- | --- | --- |
| **Modulo 4** : La gestione della memoria secondaria | | | |
| **Prerequisiti:** I moduli precedenti | | | |
| **Obiettivo generale dell’unità**: Trattare la gestione dei file in C sottolineando la differenza tra file di testo e binari e rendere consapevoli gli studenti delle problematiche legate alla allocazione dinamica in memoria. | | | |
| 1. **Obiettivi** | 1. **Contenuti** | 1. **Metodi** | 1. **Periodo** |
| La memoria secondaria e l'allocazione dinamica della memoria centrale. | * File di testo e binari * Funzioni di scrittura e lettura da file binari e di testi, fread, fwrite, fscang, fprintf, fgets, fgetc * Principali funzioni per I/O, apertura, chiusura, scrittura e lettura * Operatore sizeof, matrici e vettori come puntatori, l'aritmetica dei puntatori in C. | * Lezione frontale * Lavoro in laboratorio su piccoli progetti | Maggio/Giugno |