

Programmazione disciplinare: Informatica

Liceo scientifico - Opzione Scienze Applicate

(Anno scolastico 2025/2026)

Linee generali e competenze

L'insegnamento dell'informatica deve contemperare diversi obiettivi: comprendere i principali fondamenti teorici delle scienze dell'informazione, acquisire la padronanza di strumenti dell'informatica, utilizzare tali strumenti per la soluzione di problemi significativi in generale, ma in particolare connessi allo studio della altre discipline, acquisire la consapevolezza dei vantaggi e dei limiti dell'uso degli strumenti e dei metodi informatici e delle conseguenze sociali e culturali di tale uso.

Al termine del percorso liceale lo studente dovrà acquisire la conoscenza e la padronanza dei più comuni strumenti di software per il calcolo, la ricerca e la comunicazione in rete, la comunicazione multimediale, l'acquisizione e l'organizzazione dei dati applicandoli in una vasta gamma di situazioni, ma soprattutto nell'indagine scientifica, scegliendo di volta in volta lo strumento più adatto.

Verranno proposti problemi significativi che consentano un collegamento tra l'informatica e le altre discipline allo scopo di far acquisire al discente un ulteriore strumento di lavoro.

Il discente dovrà essere consapevole delle ragioni che hanno prodotto lo sviluppo scientifico e tecnologico nel tempo in relazione ai bisogni e alle domande di conoscenza dei diversi contesti, ponendo attenzione alle conquiste scientifiche, in particolare quelle più recenti.

L'allievo dovrà anche comprendere il ruolo della tecnologia, come mediazione tra scienza e vita quotidiana e saper utilizzare gli strumenti informatici in relazione all'analisi dei dati e alla modellizzazione di specifici problemi scientifici.

Gli obiettivi didattici ed educativi possono essere pertanto espressi genericamente come segue:

- Acquisire un metodo di studio autonomo e flessibile, che consenta di condurre ricerche e approfondimenti personali
- Essere consapevoli della diversità dei metodi utilizzati ed essere in grado valutare i criteri di affidabilità dei risultati in essi raggiunti
- Saper compiere le necessarie interconnessioni tra i vari metodi ed i vari contenuti
- Acquisire l'abitudine a ragionare con rigore logico, ad identificare i problemi e a individuare possibili soluzioni
- Saper collocare storicamente lo sviluppo delle varie invenzioni tecnologiche
- Essere in grado di utilizzare criticamente strumenti informatici e telematici nelle attività di studio e di approfondimento
- Comprendere la valenza metodologica dell'informatica nella formalizzazione e modellizzazione dei processi complessi e nell'individuazione di procedimenti risolutivi
- Acquisire padronanza del linguaggio tecnico, logico e formale della disciplina
- Saper utilizzare strumenti di calcolo e di rappresentazione per la modellizzazione e la risoluzione di problemi
- Essere consapevoli delle ragioni che hanno prodotto lo sviluppo scientifico e tecnologico nel tempo, in relazione ai bisogni e alle domande di conoscenza dei diversi contesti, con attenzione critica alle dimensioni tecnico-applicative ed etiche delle conquiste scientifiche, in particolare quelle più recenti
- Saper cogliere la potenzialità delle applicazioni e delle invenzioni tecnologiche nella vita quotidiana
- Comprendere il ruolo della tecnologia come mediazione fra scienza e vita quotidiana

- Saper utilizzare gli strumenti informatici in relazione all'analisi dei dati e alla modellizzazione di specifici problemi scientifici ed individuare la funzione dell'informatica nello sviluppo scientifico
- Acquisire la consapevolezza dei vantaggi e dei limiti dell'uso degli strumenti e dei metodi informatici e delle conseguenze sociali e culturali di tale uso
- Comprendere la struttura logico-funzionale della struttura fisica e del software di un computer e di reti locali, tale da consentirgli la scelta dei componenti più adatti alle diverse situazioni e le loro configurazioni, la valutazione delle prestazioni, il mantenimento dell'efficienza
- Saper collegare in modo sistematico l'uso di strumenti e la creazione di applicazioni ai concetti teorici ad essi sottostanti

La disciplina sarà trattata con varie metodologie didattiche:

- a) Lezioni frontali: il docente descrive con l'aiuto degli strumenti disponibili (lavagna, computer, videoproiettore, dispense, libri di testo in adozione, ecc) gli aspetti importanti dell'argomento trattato, non limitandosi alla semplice esposizione, ma stimolando la partecipazione costruttiva della classe e privilegiando il metodo deduttivo.
- b) Discussione in classe: si creano situazioni di confronto su tematiche inerenti gli argomenti trattati al fine di far emergere problemi, dubbi e congetture utili al rafforzamento dell'azione formativa.
- c) Esercitazioni pratiche e in laboratorio: dopo aver illustrato gli aspetti teorici dell'argomento, viene assegnato agli allievi un lavoro di progettazione e realizzazione. Grazie all'attività di laboratorio vengono messe alla prova le abilità progettuali e organizzative acquisite.
- d) Lavoro di gruppo al fine di stimolare la cooperazione e il confronto.
- e) Nel caso di insorgenza di problemi legati a situazioni emergenziali, per l'implementazione della DDI ci si propone di attenersi a quanto segue: ciascun docente sceglierà in corso d'opera in base al contesto in cui si troverà ad operare la tipologia di attività sincrona o asincrona da svolgere

Valutazione (tipologia di prove, criteri di valutazione e griglie)

Per quanto riguarda i compiti scritti si terrà conto della completezza dell'elaborato, della strategia risolutiva, del calcolo e dell'esposizione formale.

Potranno essere effettuate prove tra le seguenti tipologie a seconda delle esigenze e delle strategie necessarie per il consolidamento degli argomenti proposti:

- 1) Prove strutturate su una o più unità didattiche
- 2) Prove semistrutturate su una o più unità didattiche

Per quanto riguarda le prove strutturate, la griglia di valutazione che verrà adottata prevederà l'attribuzione di un punteggio ad ogni quesito. La somma totale sarà poi normalizzata e riportata in decimi.

Anche per quanto riguarda le prove semistrutturate, la griglia di valutazione che verrà adottata prevederà l'attribuzione di un punteggio ad ogni quesito. Tuttavia per i quesiti che prevedono una risposta articolata e dunque la valutazione di diversi aspetti, il punteggio verrà attribuito in base al grado di conoscenza dimostrato, in base alle competenze in termini di analisi, sintesi e deduzione ed in base alla chiarezza espositiva. Dunque nell'attribuzione del punteggio per tali quesiti ci si potrà basare sulla griglia di valutazione. Il tutto però sarà seguito da una successiva normalizzazione del punteggio. In tal caso il punteggio atto a raggiungere un voto minimo pari a 1 che scaturisce dai primi due indicatori potrà essere aggiunto al computo finale normalizzato piuttosto che ad ogni singolo quesito per evitare di rendere il punteggio del singolo quesito di difficile interpretazione da parte dell'alunno. In ogni caso nell'attribuzione del punteggio ad ogni quesito si potrà sempre tener conto del peso di ogni indicatore così come indicato nella griglia.

Tuttavia ci si riserva la possibilità di inserire nelle prove quesiti a risposte chiusa o multipla. Ovviamente in tal caso l'attribuzione del punteggio non seguirà la scala della griglia di valutazione, ma seguirà gli standard delle risposte vero/falso.

Nelle sezioni Cambridge verranno svolte simulazioni d'esame IGCSE e Advanced con attribuzione di punteggio in linea con quanto previsto dalle prove d'esame Cambridge.

Ci si riserva anche la possibilità di effettuare dei colloqui orali. Tali colloqui verranno valutati non solo per quanto riguarda la conoscenza, la comprensione degli argomenti e le competenze dimostrate, ma anche per la chiarezza dell'esposizione.

Complessivamente ci saranno almeno due valutazioni nel trimestre e tre valutazioni nel pentamestre.

Il docente potrà decidere di attribuire pesi diversi ad alcune prove, ma solo dopo averlo comunicato preventivamente agli studenti.

Si allega di seguito la griglia di valutazione.

Per la valutazione finale di ciascun alunno si terrà conto:

- della valutazione dei compiti scritti, dei colloqui orali e di altre prove somministrate nell'ambito di eventuale DDI;
- delle competenze informatiche acquisite;
- del comportamento globale (presenza, attenzione, partecipazione, impegno, correttezza in classe ed in laboratorio);
- della costanza nel lavoro autonomo;
- della puntualità delle consegne;
- del progresso rispetto ai livelli di partenza;
- dell'autonomia di lavoro;
- della partecipazione in eventuali attività di DDI

Griglia di valutazione

Indicatori	Descrittori	Giudizio sintetico	PUNTEGGIO
CONOSCENZE			
Dell'argomento e del contesto	Non risponde a quanto richiesto	Nulle	0.5
	Risponde solo ad alcune richieste e in modo approssimato	Gravemente insufficienti	1 - 2
	Risponde solo ad alcune richieste	Insufficienti	2.5 - 3.5
	Risponde alle richieste in modo essenziale	Sufficienti	4
	Risponde a tutte le richieste, in alcuni casi, in modo esauriente	Buone	4.5-5
	Risponde a tutte le richieste in modo esauriente e personalizzato	Ottime	5.5-6
COMPETENZE			

Analisi, sintesi, deduzione	Non riesce ad interpretare il quesito proposto	Nulle o gravemente insufficienti	0.5
	Confonde i concetti fondamentali	Insufficienti	1-1.5
	Coglie il significato essenziale delle informazioni	Sufficienti	2
	Riorganizza ed elabora i dati e i concetti da cui sa trarre deduzioni logiche	Buone o ottime	2.5-3
Chiarezza della soluzione e/o originalità della soluzione (da prendere in considerazione solo se si è riportata almeno la sufficienza nei punti precedenti)	Non chiara	Non presente	0
	Chiara e/o originale	Presente	0.5-1

Le eventuali verifiche degli apprendimenti potranno essere valutate utilizzando una griglia di valutazione che tenga conto di criteri relativi alle conoscenze, competenze e chiarezza/originalità della soluzione esposti nella griglia qui riportata; tuttavia tali criteri potranno essere declinati in modo differente a seconda della tipologia della prova. In ogni caso i criteri di valutazione saranno comunicati ai ragazzi prima dello svolgimento della prova. Ciò potrà avvenire anche per lavori che verranno svolti in autonomia in modalità asincrona, in modo individuale o in gruppo, che potranno prevedere o meno la realizzazione di un prodotto multimediale.

Attività di recupero

Verranno effettuate attività di recupero all’inizio del pentamestre nella settimana della pausa didattica, nel caso in cui la scuola dovesse deliberare in tal senso. Inoltre, se necessario, verranno svolte ulteriori attività di recupero in itinere per consentire agli allievi un più agevole approccio alla disciplina.

Tempi e scansione dei contenuti

Si fa riferimento a quanto indicato nella organizzazione modulare che segue.
 Ci si riserva la possibilità di svolgere i medesimi contenuti con una scansione non sequenziale.

Rubrica di valutazione delle competenze

Si allega di seguito la rubrica di valutazione delle seguenti competenze:

- Analisi del problema
- Ricerca della soluzione
- Implementazione della soluzione

COMPETENZA	DESCRIZIONE	INDICATORE	LIVELLO
ANALISI DEL PROBLEMA	<ul style="list-style-type: none">• Saper identificare gli elementi essenziali del problema e/o dell'argomento proposto• Saper riconoscere, riorganizzare e rielaborare i dati disponibili• Saper identificare gli obiettivi richiesti• Saper esporre in maniera chiara e con linguaggio appropriato i contenuti relativi al problema e/o all'argomento proposto	Sa analizzare il problema e/o l'argomento proposto in modo autonomo e funzionale al contesto, individuando input/output e obiettivi ed esponendo i contenuti richiesti in maniera chiara ed esauriente	AVANZATO
		Sa analizzare il problema e/o l'argomento proposto in modo adeguato e funzionale al contesto, individuando input/output e obiettivi ed esponendo i contenuti richiesti in maniera adeguata	INTERMEDIO
		Sa identificare in modo essenziale gli elementi base del problema e/o dell'argomento proposto, individuando input/output e obiettivi ed esponendo i contenuti richiesti in maniera sufficiente	BASE
		Non riesce ad analizzare il problema e/o l'argomento proposto nei suoi tratti essenziali nè ad individuare input/output e obiettivi. Espone i contenuti in maniera non chiara e approssimativa	NON RAGGIUNTO
RICERCA DELLA SOLUZIONE	<ul style="list-style-type: none">• Saper applicare le conoscenze al fine di elaborare strategie di risoluzione del problema• Saper individuare la soluzione più efficace/efficiente• Saper descrivere la soluzione scelta	Sa identificare e formalizzare la soluzione con un linguaggio appropriato e in modo autonomo e funzionale al contesto, individuando la soluzione ottimale	AVANZATO
		Sa identificare e formalizzare con un linguaggio appropriato e in modo adeguato la soluzione	INTERMEDIO
		Sa identificare e formalizzare con un linguaggio appropriato e in modo essenziale la soluzione	BASE
		Non riesce ad identificare e formalizzare con un linguaggio appropriato la soluzione al problema	NON RAGGIUNTO

IMPLEMENTAZIONE DELLA SOLUZIONE	<ul style="list-style-type: none"> • Saper implementare la soluzione sfruttando le conoscenze acquisite • Saper correggere gli errori 	Sa implementare la soluzione in modo autonomo, scegliendo gli strumenti più idonei alla soluzione individuata. Inoltre è in grado di rielaborare la soluzione alla luce dei problemi riscontrati	AVANZATO
		Sa implementare la soluzione in modo autonomo. Inoltre è in grado di rielaborare la soluzione alla luce dei problemi riscontrati	INTERMEDIO
		Sa implementare una possibile soluzione in modo essenziale	BASE
		Non riesce ad implementare senza errori la soluzione al problema	NON RAGGIUNTO

L'analisi di un problema, la ricerca e l'implementazione della soluzione di un problema costituiscono gli elementi fondanti della disciplina, pertanto il dipartimento ritiene che il livello complessivo raggiunto possa essere considerato sufficiente solo in presenza di livelli superiori o uguali a quello base in tutti e tre i descrittori.

In particolare, l'implementazione della soluzione è un elemento imprescindibile; pertanto il raggiungimento del livello base nell'analisi e nella ricerca della soluzione non implica un livello di competenza complessivamente sufficiente.

Le prove di verifica o singoli quesiti facenti parte delle prove di verifica potranno essere volti a testare tutte le competenze previste nella rubrica o anche solo una o due di esse

Organizzazione modulare dell'insegnamento di Informatica – Classe Prima – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di problem solving volte all'analisi del livello della classe e allo sviluppo del pensiero computazionale	14
A	Architettura del computer, rappresentazione dell'informazione	16
B	Operazioni logiche sui dati	10
C	Algoritmi e diagrammi di flusso	16
D	Introduzione alla programmazione	10
	Totale	66

Organizzazione didattica del modulo A:

Prerequisiti
- Conoscere e saper applicare le operazioni aritmetiche di base

Obiettivi
<ul style="list-style-type: none">- Conoscere le caratteristiche architettoniche di un computer- Conoscere i concetti di hardware e software- Conoscere il sistema di numerazione binario- Conoscere l'architettura di von Neumann- Conoscere il concetto di sistema operativo, le sue funzionalità di base e le caratteristiche dei sistemi operativi più comuni- Saper inquadrare la disciplina da un punto di vista storico- Saper rappresentare i dati con i numeri binari- Saper operare con numeri binari- Saper convertire numeri decimali in basi diverse- Comprendere ed utilizzare le tecniche per la rappresentazione dei dati all'interno di un computer

- Saper codificare l'informazione
- Saper decodificare l'informazione

Contenuti del modulo A:

Aspetti storici

Il computer

L'unità centrale di elaborazione (CPU)

La memoria RAM, la memoria ROM

La memoria di massa

Le periferiche di input e le periferiche di output

I sistemi posizionali

Il sistema binario

Aritmetica binaria

Il sistema esadecimale

Rappresentazione in complemento a due

Rappresentazione in virgola mobile

Codifica e decodifica dell'informazione

Organizzazione didattica del modulo B:

Prerequisiti
- Conoscere e saper applicare le operazioni aritmetiche di base

Obiettivi
- Conoscere le operazioni logiche fondamentali
- Saper effettuare l'analisi di un semplice circuito logico
- Saper effettuare la sintesi di un semplice circuito logico

Contenuti del modulo B:

Le operazioni logiche

Gli operatori logici: AND, OR, NOT, XOR, NAND, NOR

Circuiti logici

Circuiti logici a più livelli

Organizzazione didattica del modulo C:

Prerequisiti
- Conoscere le caratteristiche architetture di un computer

Obiettivi
- Saper realizzare il diagramma di flusso di un algoritmo che risolva semplici problemi usando le strutture di controllo di sequenza, selezione e iterazione

Contenuti del modulo C:

- Gli algoritmi
- I diagrammi di flusso
- Le strutture di controllo
- La sequenza
- La selezione
- L'iterazione

Organizzazione didattica del modulo D:

Prerequisiti
- Conoscere il concetto di algoritmo
- Conoscere la struttura di un computer
- Conoscere le fasi di un processo logico/tecnologico per giungere alla risoluzione di un problema in ottica informatica
- Conoscere le metodologie per la stesura di un algoritmo
- Conoscere le caratteristiche fondamentali delle istruzioni che compongono un algoritmo

- | |
|--|
| - Saper rappresentare algoritmi mediante diagrammi di flusso |
|--|

Obiettivi
<ul style="list-style-type: none">- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma- Saper codificare un algoritmo in un linguaggio di programmazione- Conoscere i principi della programmazione strutturata per costruire programmi ben ordinati attraverso le strutture di sequenza, selezione ed iterazione

Contenuti del modulo D:

Le basi del linguaggio C/C++

Le regole sintattiche

Introduzione alle strutture di controllo

Esempi di programmazione di semplici algoritmi

Obiettivi minimi

- Conoscere le caratteristiche architettoniche di un computer
- Saper rappresentare i dati con i numeri binari
- Saper effettuare conversioni tra sistema binario, sistema decimale e sistema esadecimale
- Conoscere le operazioni logiche di base
- Saper realizzare il diagramma di flusso di un algoritmo che risolva semplici problemi usando le strutture di controllo di sequenza, selezione e iterazione
- Saper effettuare l'analisi di un semplice circuito logico
- Saper realizzare semplici programmi in C/C++ con istruzioni di base

Organizzazione modulare dell'insegnamento di Informatica – Classe Seconda – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o problemi di carattere logico e algoritmico	8
A	Dal problema al processo risolutivo: gli algoritmi e la programmazione: istruzioni di base di C++	22
B	La programmazione: array e sottoprogrammi in C++	26
C	La programmazione: le stringhe in C++	10
	Totale	66

Organizzazione didattica del modulo 0:

Obiettivi
- Sviluppare il pensiero computazionale

Contenuti del modulo 0:

Problem solving
Elaborazione e confronto di strategie
Analisi dei casi possibili

Organizzazione didattica del modulo A:

Prerequisiti

- | |
|---|
| <ul style="list-style-type: none">- Conoscere il concetto di algoritmo- Conoscere il concetto di linguaggio di programmazione- Conoscere la struttura di un computer- Conoscere le fasi di un processo logico/tecnologico per giungere alla risoluzione di un problema in ottica informatica- Conoscere le metodologie per la stesura di un algoritmo- Conoscere le caratteristiche fondamentali delle istruzioni che compongono un algoritmo- Saper rappresentare algoritmi mediante diagrammi di flusso |
|---|

Obiettivi

- | |
|---|
| <ul style="list-style-type: none">- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma- Saper codificare un algoritmo in un linguaggio di programmazione- Conoscere i principi della programmazione strutturata per costruire programmi ben ordinati attraverso le strutture di sequenza, selezione ed iterazione |
|---|

Contenuti del modulo A:

Le basi del linguaggio C++
Dichiarazione di variabili
L'assegnazione dei valori alle variabili
Istruzioni di input e output
Istruzioni di selezione (if else switch)
Istruzioni di iterazione (for, while, do while)

Organizzazione didattica del modulo B:

Prerequisiti

- | |
|--|
| <ul style="list-style-type: none">- Conoscere le strutture di selezione e controllo- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma- Saper rappresentare algoritmi mediante diagrammi di flusso- Saper codificare un algoritmo in un linguaggio di programmazione |
|--|

Obiettivi
<ul style="list-style-type: none">- Comprendere l'importanza della modularità in programmazione- Saper realizzare programmi che utilizzino sottoprogrammi- Saper ricorrere all'utilizzo di array monodimensionali e bidimensionali

Contenuti del modulo B:

I sottoprogrammi
Procedure e funzioni
Gli array monodimensionali
Matrici

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Saper rappresentare algoritmi mediante diagrammi di flusso- Saper codificare un algoritmo in un linguaggio di programmazione

Obiettivi
<ul style="list-style-type: none">- Saper effettuare operazioni di input e output con le stringhe- Saper concatenare due stringhe- Saper verificare se una stringa è vuota o meno- Saper accedere ad un singolo elemento di una stringa- Saper calcolare la lunghezza di una stringa- Saper inserire una stringa all'interno di un'altra- Saper estrarre, cancellare o sostituire una sottostringa- Saper individuare una sottostringa all'interno di una stringa

Contenuti del modulo C:

Il tipo string

Dichiarazione ed inizializzazione di una stringa

L'operatore concatenazione

I metodi per la manipolazione delle stringhe

Obiettivi minimi

- Saper elaborare una strategia risolutiva in semplici casi di problem solving
- Conoscere i principi della programmazione strutturata per costruire programmi attraverso le strutture di sequenza, selezione ed iterazione
- Comprendere l'importanza della modularità in programmazione
- Saper realizzare in C++ programmi che utilizzino sottoprogrammi
- Saper effettuare in C++ operazioni elementari con gli array (lettura, scrittura, ricerca di un elemento,...) e con le stringhe

Organizzazione modulare dell'insegnamento di Informatica – Classe Terza – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e potenziamento delle capacità logiche	8
A	La programmazione: algoritmi avanzati	10
B	La programmazione: input ed output da file di testo	18
C	Puntatori e strutture dinamiche	30
	Totale	66

Obiettivi
- Sviluppare il pensiero computazionale

Contenuti del modulo 0:

Problem solving
Elaborazione e confronto di strategie
Analisi dei casi possibili

Organizzazione didattica del modulo A:

Prerequisiti
- Saper codificare un algoritmo in un linguaggio di programmazione
- Saper gestire la programmazione modulare

Obiettivi
- Saper realizzare programmi che implementino algoritmi avanzati

Contenuti del modulo A:

Generalità su algoritmi avanzati
Risoluzione problemi tratti dalle olimpiadi di informatica

Organizzazione didattica del modulo B:

Prerequisiti
<ul style="list-style-type: none">- Saper rappresentare algoritmi mediante diagrammi di flusso- Saper codificare un algoritmo in un linguaggio di programmazione
Obiettivi
<ul style="list-style-type: none">- Saper effettuare operazioni di input e output da file

Contenuti del modulo B:

Archiviazione di dati su file
Istruzioni per l'apertura e chiusura dei file
Istruzioni per la scrittura e lettura dei file

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Conoscere la rappresentazione della memoria di un calcolatore- Conoscenze gli elementi di base del linguaggio C++
Obiettivi
<ul style="list-style-type: none">- Conoscere le caratteristiche di struct, puntatori, liste, pile, code- Saper implementare in C++ le strutture dinamiche dei dati- Saper creare una lista- Saper visualizzare una lista

- Saper inserire un nuovo nodo in una lista in testa, in coda ed in una posizione specifica
- Saper cancellare un nodo da una lista
- Saper creare una pila
- Saper visualizzare una pila
- Saper inserire un nuovo nodo in una pila
- Saper estrarre un nodo da una pila
- Saper eliminare una pila
- Saper creare una coda
- Saper visualizzare una coda
- Saper inserire un nuovo nodo in una coda
- Saper estrarre un nodo da una coda
- Saper eliminare una coda

Contenuti del modulo C:

I record

I puntatori

Le liste: creazione di una lista, visualizzazione di una lista, inserimento di nuovi nodi in una lista in testa, in coda ed in una posizione specifica, cancellazione di un nodo da una lista

Le pile: creazione di una pila, visualizzazione di una pila, inserimento di un nodo in una pila, estrazione di un nodo da una pila, eliminazione di una pila

Le code: creazione di una coda, visualizzazione di una coda, inserimento di un nodo in una coda, estrazione di un nodo da una coda, eliminazione di una coda

Obiettivi minimi

- Saper effettuare operazioni di input e output da file
- Conoscere le caratteristiche di struct, puntatori, liste, pile, code
- Saper implementare in C++ le strutture dinamiche dei dati
- Saper effettuare operazioni di base sulle strutture dinamiche
- Saper dare una descrizione concettuale degli algoritmi avanzati illustrati in classe

Organizzazione modulare dell'insegnamento di Informatica – Classe Quarta – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e potenziamento delle capacità logiche	5
A	Elementi di basi di dati	12
B	Pagine web	12
C	Java: elementi di base, strutture di controllo, gestione I/O, array e matrici	10
D	Java: Object Oriented. Incapsulamento, Ereditarietà, Polimorfismo	16
E	Riflessioni sul ruolo della tecnologia nella storia e nella società (educazione civica)	11
	Totale	66

Organizzazione didattica del modulo 0:

Obiettivi
- Sviluppare il pensiero computazionale

Contenuti del modulo 0:

Problem solving
Elaborazione e confronto di strategie
Analisi dei casi possibili

Organizzazione didattica del modulo A:

Prerequisiti
- HW e SW (di base e applicativo)

Obiettivi
- Individuare la necessità di memorizzare dati e documenti

- | |
|---|
| <ul style="list-style-type: none">- Riconoscere le caratteristiche dei dati- Riconoscere le funzioni di un database- Utilizzare le funzioni per estrarre informazioni da una base di dati- Saper progettare il modello concettuale di un database- Saper derivare il modello logico dal modello concettuale |
|---|

Contenuti del modulo A:

Le basi di dati
Le fasi della progettazione
Il DBMS e le sue funzioni
Il modello E/R: entità e attributi
Il modello E/R: le associazioni
I modelli logici: modello relazionale
Il modello fisico
I vincoli
Interrogare una base di dati (query)

Organizzazione didattica del modulo B:

Prerequisiti
<ul style="list-style-type: none">- Conoscere i concetti fondamentali relativi alla comunicazione in rete

Obiettivi
<ul style="list-style-type: none">- Saper creare una pagina web- Saper realizzare il layout di una pagina web- Saper realizzare un form riuscendo a disporre i vari elementi nelle diverse zone della pagina web

Contenuti del modulo B:

Il linguaggio html

I tag fondamentali per la creazione di una pagina web
I tag di formattazione
Inserimento di immagini
Il layout di una pagina web
Elenchi, link e tabelle
I form
Esempi di programmazione di pagine web

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Conoscere il concetto di algoritmo- Conoscere il concetto di linguaggio di programmazione- Conoscere la struttura di un computer- Conoscere le fasi di un processo logico/tecnologico per giungere alla risoluzione di un problema in ottica informatica- Conoscere le metodologie per la stesura di un algoritmo- Conoscere le caratteristiche fondamentali delle istruzioni che compongono un algoritmo- Saper rappresentare algoritmi mediante diagrammi di flusso
Obiettivi
<ul style="list-style-type: none">- Saper scrivere i programmi utilizzando in modo corretto la sintassi del linguaggio- Riconoscere le diverse fasi del lavoro di programmazione per codificare e validare gli algoritmi

Contenuti del modulo C:

Caratteristiche generali
L'ambiente di programmazione
La struttura dei programmi
Identificatori, parole chiave, variabili e costanti
La gestione dell'input/output
Le strutture di controllo: sequenza, selezione, ripetizione
La struttura di dati array

Organizzazione didattica del modulo D:

Prerequisiti
- Conoscere i fondamenti della programmazione del linguaggio Java

Obiettivi
- Saper definire classi con attributi e metodi - Saper creare e utilizzare oggetti - Applicare i principi della programmazione ad oggetti utilizzando il linguaggio Java.

Contenuti del modulo D:

Orientamento agli oggetti

Gli oggetti e le classi

Dichiarazione e utilizzo di una classe, degli attributi e dei metodi

Creazione e utilizzo degli oggetti

Ereditarietà

Polimorfismo

Organizzazione didattica del modulo E:

Obiettivi
- Conoscere aspetti storici relativi alla tecnologia - Saper effettuare riflessioni critiche sulle opportunità e i rischi legati alla tecnologia

Contenuti del modulo E:

La tecnologia

Opportunità legate all'uso delle nuove tecnologie

Rischi legati all'uso delle nuove tecnologie

Per quanto riguarda la valutazione relativa ad Educazione Civica il docente potrà adottare o la griglia di istituto o una propria griglia di valutazione

Obiettivi minimi

- Utilizzare le funzioni per estrarre informazioni da una base di dati
- Saper creare un semplice database
- Saper creare una pagina web
- Saper realizzare un form
- Saper scrivere i programmi utilizzando in modo corretto la sintassi del linguaggio Java
- Saper definire classi con attributi e metodi
- Saper creare e utilizzare oggetti
- Applicare i principi della programmazione ad oggetti utilizzando il linguaggio Java.

Organizzazione modulare dell'insegnamento di Informatica – Classe Quinta – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
A	Sistemi intelligenti	14
B	Realizzazione di applicazioni a carattere scientifico	18
C	Progettazione web dinamica	12
D *	Crittografia	12
E	Reti	10
	Totale	66

* Il modulo potrà esser svolto in modalità CLIL

Organizzazione didattica del modulo A:

Prerequisiti
- Conoscere l'architettura del calcolatore

Obiettivi
- Saper esprimere il motivo per cui si ricorre ai sistemi intelligenti - Saper costruire un modello - Saper passare dal modello al progetto - Saper scegliere la soluzione ottimale

Contenuti del modulo A:

Automi intelligenti
Implementazione di macchine intelligenti
Progettazione del modello
Implementazione del modello

Organizzazione didattica del modulo B:

Prerequisiti
- Conoscere i concetti fondamentali relativi all'analisi numerica

Obiettivi
- Saper creare applicazioni per la risoluzione di problemi di carattere scientifico
- Saper illustrare e presentare con un linguaggio tecnicamente appropriato le applicazioni realizzate

Contenuti del modulo B:

Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo di bisezione
Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo di Newton
Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo delle secanti
Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo della regula falsi
Ulteriori applicazioni a carattere scientifico / matematico

Organizzazione didattica del modulo C:

Prerequisiti
- Conoscere i concetti fondamentali relativi alla comunicazione in rete

Obiettivi
- Saper creare una pagina web
- Saper realizzare il layout di una pagina web
- Saper realizzare pagine web con pagine dinamiche

Richiamo di istruzioni base html
Prelevamento di dati da form
Elaborazione di dati prelevati da form
Programmazione dinamica
Realizzazione di applicazioni web

Organizzazione didattica del modulo D:

Prerequisiti
<ul style="list-style-type: none">- Acquisire i termini tecnici relativi alla crittografia- Saper applicare alcuni algoritmi elementari a chiave simmetrica per criptare un plaintext- Saper applicare alcuni algoritmi elementari a chiave simmetrica per decriptare un ciphertext- Conoscere le motivazioni che hanno reso indispensabile l'introduzione di algoritmi a chiave asimmetrica- Saper descrivere i meccanismi su cui si basano gli algoritmi a chiave asimmetrica

Obiettivi
<ul style="list-style-type: none">- Saper effettuare codifiche e decodifiche mediante cifrario di Cesare, ROT13, cifrario di Vigenere, cifrario di Bellaso- Conoscere le differenze tra un cifrario monoalfabetico ed un cifrario polialfabetico- Conoscere caratteristiche e differenze tra algoritmi a chiave simmetrica e algoritmi a chiave asimmetrica- Saper illustrare i meccanismi che regolano gli algoritmi di crittografia moderna- Saper descrivere il funzionamento di algoritmi di crittografia moderna

Contenuti del modulo D:

Aspetti storici relativi alla crittografia
Cifrari monoalfabetici e polialfabetici
Cifrario di Cesare, ROT13
Cifrario di Vigenere
Cifrario di Bellaso
Algoritmi a chiave simmetrica e asimmetrica
Enigma e la sua decifrazione
Algoritmi di crittografia moderna

Organizzazione didattica del modulo E:

Prerequisiti
<ul style="list-style-type: none">- Conoscere i concetti fondamentali relativi alla comunicazione in rete

Obiettivi
<ul style="list-style-type: none">- Saper descrivere i principali mezzi trasmissivi- Saper illustrare e comparare le diverse topologie di rete- Saper esporre il modello ISO – OSI e l'architettura TCP/IP- Saper descrivere i diversi protocolli di rete

Contenuti del modulo E:

Mezzi trasmissivi. Topologia. Dispositivi di rete

Il modello ISO-OSI e l'architettura TCP/IP

I protocolli di rete: FTP, HTTP, HTTPS, SMTP, POP3

Obiettivi minimi

- Saper illustrare e presentare con un linguaggio tecnicamente appropriato applicazioni per la risoluzione di problemi di carattere scientifico
- Saper esprimere il motivo per cui si ricorre ai sistemi intelligenti
- Saper effettuare semplici progettazioni di sistemi intelligenti
- Saper illustrare i concetti base alla progettazione di sistemi intelligenti
- Saper creare una pagina web
- Saper realizzare il layout di una pagina web
- Saper realizzare pagine web con pagine dinamiche
- Saper illustrare i meccanismi che regolano gli algoritmi di crittografia moderna
- Saper descrivere gli algoritmi di crittografia moderna
- Saper descrivere i principali mezzi trasmissivi , le diverse topologie di rete ed i principali dispositivi di rete
- Saper illustrare il modello ISO – OSI e l'architettura TCP/IP
- Saper descrivere i diversi protocolli di rete

Organizzazione modulare dell'insegnamento di Informatica – Classe Prima Potenziamento Informatica – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di problem solving volte all'analisi del livello della classe e allo sviluppo del pensiero computazionale	22
A	Architettura del computer, rappresentazione dell'informazione	16
B	Operazioni logiche sui dati	10
C	Dal problema al processo risolutivo: gli algoritmi La programmazione: istruzioni di base di C++	26
D	La programmazione: array e sottoprogrammi in C++	20
E	La programmazione: le stringhe in C++	12
F	La programmazione: la ricorsione	16
G	Introduzione alle reti	10
	Totale	132

Organizzazione didattica del modulo A:

Prerequisiti
- Conoscere e saper applicare le operazioni aritmetiche di base

Obiettivi
- Conoscere le caratteristiche architetture di un computer
- Conoscere i concetti di hardware e software
- Conoscere il sistema di numerazione binario
- Conoscere l'architettura di von Neumann
- Conoscere il concetto di sistema operativo, le sue funzionalità di base e le caratteristiche dei sistemi operativi più comuni

- Conoscere come avviene la trasmissione dei dati e problemi relativi alla sicurezza
- Saper inquadrare la disciplina da un punto di vista storico
- Saper rappresentare i dati con i numeri binari
- Saper operare con numeri binari
- Saper convertire numeri decimali in basi diverse
- Comprendere ed utilizzare le tecniche per la rappresentazione dei dati all'interno di un computer
- Saper codificare l'informazione
- Saper decodificare l'informazione

Contenuti del modulo A:

Aspetti storici

Il computer

L'unità centrale di elaborazione (CPU)

La memoria RAM, la memoria ROM

La memoria di massa

Le periferiche di input e le periferiche di output

I sistemi posizionali

Il sistema binario

Aritmetica binaria

Il sistema esadecimale

Rappresentazione in complemento a due

Rappresentazione in virgola mobile

Codifica e decodifica dell'informazione

Organizzazione didattica del modulo B:

Prerequisiti

- Conoscere e saper applicare le operazioni aritmetiche di base

Obiettivi

- | |
|---|
| <ul style="list-style-type: none">- Conoscere le operazioni logiche fondamentali- Saper effettuare l'analisi di un semplice circuito logico- Saper effettuare la sintesi di un semplice circuito logico |
|---|

Contenuti del modulo B:

Le operazioni logiche

Gli operatori logici: AND, OR, NOT, XOR, NAND, NOR

Circuiti logici

Circuiti logici a più livelli

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Conoscere le caratteristiche architetture di un computer

Obiettivi
<ul style="list-style-type: none">- Saper realizzare il diagramma di flusso di un algoritmo- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma- Saper codificare un algoritmo in un linguaggio di programmazione- Conoscere i principi della programmazione strutturata per costruire programmi ben ordinati attraverso le strutture di sequenza, selezione ed iterazione

Contenuti del modulo C:

Gli algoritmi

I diagrammi di flusso

Le strutture di controllo

La sequenza

La selezione

L'iterazione
Le basi del linguaggio C++
Dichiarazione di variabili
L'assegnazione dei valori alle variabili
Istruzioni di input e output
Istruzioni di selezione (if else switch)
Istruzioni di iterazione (for, while, do while)

Organizzazione didattica del modulo D:

Prerequisiti
<ul style="list-style-type: none">- Conoscere le strutture di selezione e controllo- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma- Saper rappresentare algoritmi mediante diagrammi di flusso- Saper codificare un algoritmo in un linguaggio di programmazione

Obiettivi
<ul style="list-style-type: none">- Comprendere l'importanza della modularità in programmazione- Saper realizzare programmi che utilizzino sottoprogrammi- Saper gestire il passaggio di parametri- Saper ricorrere all'utilizzo di array monodimensionali e bidimensionali

Contenuti del modulo D:

I sottoprogrammi
Sottoprogrammi con parametri
Il passaggio di parametri
Procedure e funzioni
Gli array

Organizzazione didattica del modulo E:

Prerequisiti
<ul style="list-style-type: none">- Saper rappresentare algoritmi mediante diagrammi di flusso- Saper codificare un algoritmo in un linguaggio di programmazione

Obiettivi
<ul style="list-style-type: none">- Saper effettuare operazioni di input e output con le stringhe- Saper concatenare due stringhe- Saper verificare se una stringa è vuota o meno- Saper accedere ad un singolo elemento di una stringa- Saper calcolare la lunghezza di una stringa- Saper inserire una stringa all'interno di un'altra- Saper estrarre, cancellare o sostituire una sottostringa- Saper individuare una sottostringa all'interno di una stringa

Contenuti del modulo E:

Il tipo string

Dichiarazione ed inizializzazione di una stringa

L'operatore concatenazione

I metodi empty, At, length, insert, substr, erase, replace, find

Organizzazione didattica del modulo F:

Prerequisiti
<ul style="list-style-type: none">- Saper analizzare un problema distinguendo i dati di input e di output e le operazioni necessarie per arrivare al risultato atteso a partire dai dati disponibili- Saper ricavare un programma C++ sintatticamente corretto a partire dall'algoritmo- Saper distinguere i problemi in cui è possibile utilizzare funzioni e procedure- Saper distinguere i problemi in cui è opportuno utilizzare funzioni e procedure- Saper sviluppare un programma facendo uso di funzioni e procedure

Obiettivi
<ul style="list-style-type: none">- Sapere cosa si intende per ricorsione- Conoscere le caratteristiche della ricorsione- Saper descrivere la differenza tra ricorsione ed iterazione e saper identificare i vantaggi e gli svantaggi di entrambe- Saper determinare se di fronte ad un problema si può applicare la ricorsione, se è opportuno usarla o se è preferibile applicare l'iterazione- Saper utilizzare la tecnica della ricorsione all'interno di un programma C++

Contenuti del modulo F:

Dall'induzione matematica alla ricorsione. Principali caratteristiche della ricorsione
Occupazione di memoria nel caso di programma ricorsivo
Differenze tra ricorsione ed iterazione

Organizzazione didattica del modulo G:

Prerequisiti
<ul style="list-style-type: none">- Saper rappresentare i numeri in base 2- Saper convertire un numero binario in un numero decimale- Saper convertire un numero decimale in un numero binario

Obiettivi
<ul style="list-style-type: none">- Conoscere le caratteristiche fisiche dei mezzi trasmissivi- Saper indicare quando è opportuno ricorrere ad un mezzo trasmissivo piuttosto che ad un altro- Conoscere le caratteristiche e le funzionalità dei dispositivi di rete- Saper indicare quando è opportuno ricorrere ad un dispositivo di rete piuttosto che ad un altro- Saper classificare le reti e le modalità di trasmissione- Conoscere le caratteristiche della comunicazione in rete ed i rischi ad essa connessi- Saper inquadrare lo sviluppo delle reti da un punto di vista storico

Contenuti del modulo G:

Mezzi trasmissivi
Caratteristiche fisiche dei mezzi elettrici
Modalità di trasmissione
Dispositivi di interconnessione tra le reti
Tipi di rete
Internet
La nascita e lo sviluppo di Internet

Obiettivi minimi

- Saper elaborare una strategia risolutiva in semplici casi di problem solving
- Conoscere le caratteristiche architettoniche di un computer
- Saper rappresentare i dati con i numeri binari
- Saper effettuare conversioni tra sistema binario, sistema decimale e sistema esadecimale
- Conoscere le operazioni logiche di base
- Saper realizzare il diagramma di flusso di un algoritmo che risolva semplici problemi usando le strutture di controllo di sequenza, selezione e iterazione
- Saper effettuare l'analisi di un semplice circuito logico
- Conoscere i principi della programmazione strutturata per costruire programmi attraverso le strutture di sequenza, selezione ed iterazione
- Comprendere l'importanza della modularità in programmazione
- Saper realizzare in C++ programmi che utilizzino sottoprogrammi
- Saper effettuare in C++ operazioni elementari con gli array (lettura, scrittura, ricerca di un elemento,...) e con le stringhe
- Saper utilizzare in casi molto semplici (ad esempio calcolo di una potenza o di un fattoriale) la tecnica della ricorsione all'interno di un programma C++
- Conoscere le caratteristiche principali delle reti

Organizzazione modulare dell'insegnamento di Informatica – Classe Seconda Potenziamento Informatica – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e potenziamento delle capacità logiche	28
A	Programmazione in C++: lettura da input e output	20
B	Strutture dati: struct, puntatori, liste, pile, code	48
C	La programmazione: algoritmi avanzati	26
D	Pagine web	10
	Totale	132

Organizzazione didattica del modulo 0:

Obiettivi
- Sviluppare il pensiero computazionale

Contenuti del modulo 0:

Problem solving
Elaborazione e confronto di strategie
Analisi dei casi possibili
Algoritmi e programmi in C/C++

Organizzazione didattica del modulo A:

Prerequisiti
- Saper rappresentare algoritmi mediante diagrammi di flusso
- Saper codificare un algoritmo in un linguaggio di programmazione

Obiettivi

- | |
|---|
| - Saper effettuare operazioni di input e output |
|---|

Contenuti del modulo A:

Archiviazione di dati su file
Istruzioni per l'apertura e chiusura dei file
Istruzioni per la scrittura e lettura dei file

Organizzazione didattica del modulo B:

Prerequisiti

- | |
|---|
| - Conoscere la rappresentazione della memoria di un calcolatore
- Conoscenze gli elementi di base del linguaggio C++ |
|---|

Obiettivi

- | |
|---|
| - Conoscere le caratteristiche di struct, puntatori, liste, pile, code
- Saper implementare in C++ le strutture dinamiche dei dati
- Saper creare una lista
- Saper visualizzare una lista
- Saper inserire un nuovo nodo in una lista in testa, in coda ed in una posizione specifica
- Saper cancellare un nodo da una lista
- Saper creare una pila
- Saper visualizzare una pila
- Saper inserire un nuovo nodo in una pila
- Saper estrarre un nodo da una pila
- Saper eliminare una pila
- Saper creare una coda
- Saper visualizzare una coda
- Saper inserire un nuovo nodo in una coda
- Saper estrarre un nodo da una coda
- Saper eliminare una coda |
|---|

Contenuti del modulo B:

I record

I puntatori

Le liste: creazione di una lista, visualizzazione di una lista, inserimento di nuovi nodi in una lista in testa, in coda ed in una posizione specifica, cancellazione di un nodo da una lista

Le pile: creazione di una pila, visualizzazione di una pila, inserimento di un nodo in una pila, estrazione di un nodo da una pila, eliminazione di una pila

Le code: creazione di una coda, visualizzazione di una coda, inserimento di un nodo in una coda, estrazione di un nodo da una coda, eliminazione di una coda

Organizzazione didattica del modulo C:

Prerequisiti
- Saper codificare un algoritmo in un linguaggio di programmazione
- Saper gestire la programmazione modulare

Obiettivi
- Saper realizzare programmi che implementino algoritmi avanzati

Contenuti del modulo C:

Generalità su algoritmi avanzati

Risoluzione di problemi con algoritmi avanzati

Organizzazione didattica del modulo D:

Prerequisiti
- Conoscere i concetti fondamentali relativi alla comunicazione in rete

Obiettivi

- Saper creare una pagina web
- Saper realizzare il layout di una pagina web
- Saper disporre i vari elementi nelle diverse zone della pagina web

Contenuti del modulo D:

Il linguaggio html

I tag fondamentali per la creazione di una pagina web

I tag di formattazione

Inserimento di immagini

Il layout di una pagina web

Elenchi, link e tabelle

Esempi di programmazione di pagine web

Obiettivi minimi

- Saper effettuare operazioni di input/output da file
- Conoscere le caratteristiche di struct, puntatori, liste, pile, code
- Saper creare una lista e saper effettuare operazioni elementari su una lista
- Saper creare una pila e saper effettuare operazioni elementari su una pila
- Saper creare una coda e saper effettuare operazioni elementari su una coda
- Saper dare una descrizione concettuale degli algoritmi avanzati illustrati in classe
- Saper creare delle semplici pagine web

Organizzazione modulare dell'insegnamento di Informatica – Classe Terza Potenziamento Informatica – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o attività di problem solving ed esercizi di logica, algoritmica volti allo sviluppo del pensiero computazionale	16
A	Pagine Web	18
B	Elementi di basi di dati	26
C	Programmazione lato client	24
D	Programmazione lato server	28
E	Realizzazione applicazioni internet end-to-end	20
	Totale	132

Organizzazione didattica del modulo A:

Prerequisiti
- HW e SW (di base e applicativo) - Saper creare una pagina web

Obiettivi
- Saper realizzare applicazioni web con più pagine - Saper realizzare un form riuscendo a disporre i vari elementi nelle diverse zone della pagina web

Contenuti del modulo A:

Richiamo dei concetti fondamentali del linguaggio html
Navigazione su più pagine
I form
Gli elementi di un form

Organizzazione didattica del modulo B:

Prerequisiti
- HW e SW (di base e applicativo)

Obiettivi
- Individuare la necessità di memorizzare dati e documenti - Riconoscere le caratteristiche dei dati - Riconoscere le funzioni di un database - Utilizzare le funzioni per estrarre informazioni da una base di dati - Saper progettare il modello concettuale di un database - Saper derivare il modello logico e fisico dal modello concettuale

Contenuti del modulo B:

Le basi di dati
Le fasi della progettazione
Il DBMS e le sue funzioni
Il modello E/R: entità e attributi
Il modello E/R: le associazioni
I modelli logici: modello relazionale
Il modello fisico
I vincoli
Interrogare una base di dati (query)

Organizzazione didattica del modulo C:

Prerequisiti
- Conoscere gli elementi fondamentali del linguaggio HTML - Conoscere i form HTML

Obiettivi

- | |
|--|
| <ul style="list-style-type: none">- Comprendere la struttura logica di una pagina web- Navigare la struttura della pagina- Saper implementare logiche per la gestione degli elementi di una pagina web |
|--|

Contenuti del modulo C:

Richiamo di istruzioni base html
Struttura di una pagina web
Accesso agli elementi della pagina
Controllo degli eventi
Implementazione di logiche per la gestione degli elementi di una pagina

Organizzazione didattica del modulo D:

Prerequisiti
- Conoscere la struttura di una pagina

Obiettivi
<ul style="list-style-type: none">- Conoscere i concetti fondamentali relativi alla comunicazione in rete- Saper gestire le richieste provenienti dal client- Saper realizzare applicazioni dinamiche

Richiamo di istruzioni base html
Scripting lato server
Prelevamento di dati da form
Gestione richieste lato client
Programmazione dinamica

Organizzazione didattica del modulo E:

Prerequisiti
- Conoscere i concetti fondamentali lato client e lato server

Obiettivi
- Saper collegare i diversi livelli di un'applicazione
- Saper individuare il livello in cui implementare una determinata funzionalità
- Saper realizzare semplici applicazioni in rete

Contenuti del modulo E:

Le componenti fondamentali di un'applicazione internet
Pagine web statiche e pagine web dinamiche
Comunicazione client-server
Elaborazione di dati prelevati da form
Memorizzazione in una base dati
Realizzazione pagine contenenti dati letti da una base dati
Implementazione di semplici applicazioni end-to-end

Obiettivi minimi

- Saper creare un semplice sito web
- Saper realizzare un form
- Saper creare un semplice database
- Saper estrarre informazioni da una base dati
- Saper scrivere semplici programmi lato client per l'accesso agli elementi di una pagina
- Saper scrivere semplici programmi lato server per il prelevamento dei dati provenienti dal client
- Saper scrivere semplici programmi lato server per la memorizzazione dei dati in una tabella
- Saper scrivere semplici programmi lato server per il recupero dei dati da una tabella
- Saper scrivere semplici programmi per collegare i livelli di un'applicazione

Organizzazione modulare dell'insegnamento di Informatica – Classe Quarta Potenziamento Informatica – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o attività di problem solving volte allo sviluppo del pensiero computazionale ed esercizi di programmazione con il ricorso a strutture dati ed algoritmi acquisiti negli anni precedenti	25
A	Progettazione di applicazioni avanzate	30
B	Riflessioni sul ruolo della tecnologia nella storia e nella società (educazione civica)	11
C	Java: elementi di base, strutture di controllo, gestione I/O, array e matrici	22
D	Java: Object Oriented. Incapsulamento, Ereditarietà, Polimorfismo	22
E	Graphical User Interface	22
	Totale	132

Organizzazione didattica del modulo A:

Prerequisiti
- Conoscere i concetti fondamentali relativi all'hardware di base e al software di base

Obiettivi
- Saper effettuare un'analisi preliminare alla progettazione - Saper utilizzare tecniche top down - Saper utilizzare tecniche di stepwise refinement - Sviluppare la conoscenza della struttura di un sistema e di come le varie parti interagiscano tra di loro - Acquisire le competenze necessarie per applicare le proprie conoscenze allo sviluppo di soluzioni computer based

Contenuti del modulo A:

Analisi del problema
Acquisizione e memorizzazione dei dati
Stepwise refinement
Operazioni di testing

Organizzazione didattica del modulo B:

Obiettivi
<ul style="list-style-type: none">- Conoscere aspetti storici relativi alla tecnologia- Saper effettuare riflessioni critiche sulle opportunità e i rischi legati alla tecnologia

Contenuti del modulo B:

La tecnologia
Opportunità legate all'uso delle nuove tecnologie
Rischi legati all'uso delle nuove tecnologie

Per quanto riguarda la valutazione relativa ad Educazione Civica il docente potrà adottare o la griglia di istituto o una propria griglia di valutazione

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Conoscere il concetto di algoritmo- Conoscere il concetto di linguaggio di programmazione- Conoscere la struttura di un computer- Conoscere le fasi di un processo logico/tecnologico per giungere alla risoluzione di un problema in ottica informatica- Conoscere le metodologie per la stesura di un algoritmo- Conoscere le caratteristiche fondamentali delle istruzioni che compongono un algoritmo- Saper rappresentare algoritmi mediante diagrammi di flusso

Obiettivi
<ul style="list-style-type: none">- Saper scrivere i programmi utilizzando in modo corretto la sintassi del linguaggio Java- Riconoscere le diverse fasi del lavoro di programmazione per codificare e validare gli algoritmi

Contenuti del modulo C:

Caratteristiche generali di Java
L'ambiente di programmazione
La struttura dei programmi
Identificatori, parole chiave, variabili e costanti
La gestione dell'input/output
Le strutture di controllo: sequenza, selezione, ripetizione
La struttura di dati array

Organizzazione didattica del modulo D:

Prerequisiti
<ul style="list-style-type: none">- Conoscere i fondamenti della programmazione del linguaggio Java

Obiettivi
<ul style="list-style-type: none">- Saper definire classi con attributi e metodi- Saper creare e utilizzare oggetti- Applicare i principi della programmazione ad oggetti utilizzando il linguaggio Java.

Contenuti del modulo D:

Orientamento agli oggetti
Gli oggetti e le classi
Dichiarazione e utilizzo di una classe, degli attributi e dei metodi
Creazione e utilizzo degli oggetti
Ereditarietà

Polimorfismo

Organizzazione didattica del modulo E:

Prerequisiti
<ul style="list-style-type: none">- Conoscere i fondamenti della programmazione visuale- Conoscere il concetto di evento
Obiettivi
<ul style="list-style-type: none">- Saper realizzare applicazioni con interfacce grafiche

Contenuti del modulo E:

Accesso e programmazione degli oggetti della GUI
Controllo dell'input
Gestione degli eventi

Obiettivi minimi

- Saper scrivere i programmi utilizzando in modo corretto la sintassi del linguaggio Java
- Saper definire classi con attributi e metodi
- Saper creare e utilizzare oggetti
- Applicare i principi della programmazione ad oggetti utilizzando il linguaggio Java
- Saper realizzare semplici interfacce grafiche
- Saper illustrare rischi e benefici legati alle nuove tecnologie
- Acquisire le competenze necessarie per applicare le proprie conoscenze allo sviluppo di soluzioni computer based

Organizzazione modulare dell'insegnamento di Informatica – Classe Quinta Potenziamento Informatica – Opzione Scienze Applicate

A	Il linguaggio Python	40
B	Le librerie per il Machine Learning	32
C	Sistemi Intelligenti	34
D	Le reti	18
E	Realizzazione di applicazioni a carattere scientifico	8
		Totale 132

Organizzazione didattica del modulo A:

Prerequisiti
<ul style="list-style-type: none">• Conoscere i principi di base della programmazione procedurale e a oggetti

Obiettivi
<ul style="list-style-type: none">• Saper implementare algoritmi di base utilizzando le strutture di controllo del linguaggio• Saper creare e gestire operazioni sulle liste• Saper estrarre sotto-liste utilizzando lo slicing• Saper costruire liste applicando la list-comprehension• Saper implementare funzioni

Contenuti del modulo A:

Le basi del linguaggio e le finalità

Le regole sintattiche

Le strutture di controllo

Le stringhe

Le liste

Operazioni con le liste dinamiche

Lo slicing

La list-comprehension

Le funzioni

Esempi di applicazioni in python

Organizzazione didattica del modulo B:

Prerequisiti
<ul style="list-style-type: none">• Conoscenza del linguaggio Python

Obiettivi
<ul style="list-style-type: none">• Saper gestire e manipolare vettori con numpy• Saper leggere, pulire e gestire dati tabellari con pandas• Saper creare grafici per l'analisi dei dati con matplotlib

Contenuti del modulo B:

La libreria numpy

Gestire e manipolare i vettori con numpy

La libreria pandas

Importare e manipolare dati tabellari con pandas

La libreria matplotlib

Creare grafici con matplotlib

Altre librerie per il machine learning

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">• Conoscere le principali librerie per il machine learning

Obiettivi
<ul style="list-style-type: none">- Comprendere l'approccio e le finalità del machine learning- Saper riconoscere le diverse tipologie di algoritmi- Comprendere la metodologia per la creazione di modelli predittivi- Comprendere i metodi per derivare conoscenza dai dati- Implementare tecniche per il train & test di dataset- Saper implementare semplici algoritmi di machine learning in python

Contenuti del modulo C:

L'intelligenza artificiale e il machine learning

Tipologie di algoritmi per il machine learning

Algoritmi supervisionati e non supervisionati, regressione e classificazione

La costruzione del modello

Train & Test

Le reti neurali

Implementazione di algoritmi di machine learning in python

Esempi di utilizzo di altri algoritmi

Organizzazione didattica del modulo D:

Prerequisiti
<ul style="list-style-type: none">• Conoscere i concetti fondamentali relativi alla comunicazione in rete

Obiettivi
<ul style="list-style-type: none">• Saper esporre il modello ISO – OSI e l'architettura TCP/IP• Saper descrivere i diversi protocolli di rete• Saper esporre gli elementi che hanno portato allo sviluppo della rete• Saper implementare semplici applicazioni in rete

Contenuti del modulo D:

Il modello ISO-OSI
La suite TCP/IP
I principali protocolli di rete
Esempi di applicazioni in rete

Organizzazione didattica del modulo E:

Prerequisiti
<ul style="list-style-type: none">• Conoscere i concetti fondamentali relativi all'analisi numerica

Obiettivi
<ul style="list-style-type: none">• Saper creare applicazioni per la risoluzione di problemi di carattere scientifico• Saper illustrare e presentare con un linguaggio tecnicamente appropriato le applicazioni realizzate

Contenuti del modulo E:

Il calcolo numerico: concetti e finalità
Algoritmo per il calcolo delle radici con il metodo di Bisezione
Implementazione del metodo di Bisezione

Algoritmo per il calcolo delle radici con il metodo di Newton
Implementazione del metodo di Newton

Obiettivi minimi

- Saper costruire e manipolare semplici liste in python
- Saper utilizzare numpy per compiere semplici operazioni sugli array
- Saper utilizzare pandas per leggere e pulire semplici dataset
- Saper creare grafici di base per il machine learning
- Saper illustrare le finalità del machine learning
- Sapere riconoscere e illustrare le diverse tipologie di algoritmi per il machine learning
- Saper descrivere i principi alla base del train & test di dataset
- Saper illustrare il modello ISO – OSI e l'architettura TCP/IP
- Saper descrivere i diversi protocolli di rete
- Saper illustrare e presentare con un linguaggio tecnicamente appropriato gli algoritmi per il calcolo delle radici

Organizzazione modulare dell'insegnamento di Informatica – Classe Prima Cambridge – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di problem solving volte all'analisi del livello della classe e allo sviluppo del pensiero computazionale	15
A	Architettura del computer, rappresentazione dell'informazione	30
B	Operazioni logiche sui dati	15
C	Dal problema al processo risolutivo: gli algoritmi La programmazione: istruzioni di base di C++	39
	Totale	99

Organizzazione didattica del modulo A:

Prerequisiti
- Conoscere e saper applicare le operazioni aritmetiche di base

Obiettivi
<ul style="list-style-type: none">- Conoscere le caratteristiche architettoniche di un computer- Conoscere i concetti di hardware e software- Conoscere il sistema di numerazione binario- Conoscere l'architettura di von Neumann- Conoscere il concetto di sistema operativo, le sue funzionalità di base e le caratteristiche dei sistemi operativi più comuni- Conoscere come avviene la trasmissione dei dati e problemi relativi alla sicurezza- Saper inquadrare la disciplina da un punto di vista storico- Saper descrivere opportunità e rischi legati allo sviluppo della robotica e dell'intelligenza artificiale- Saper rappresentare i dati con i numeri binari- Saper operare con numeri binari- Saper convertire numeri decimali in basi diverse- Saper rappresentare un numero in complemento a 2

- Saper effettuare operazioni di somma e sottrazione tra numeri rappresentati in complemento a 2
- Comprendere ed utilizzare le tecniche per la rappresentazione dei dati all'interno di un computer
- Saper codificare l'informazione
- Saper decodificare l'informazione

Contenuti del modulo A:

Aspetti storici

Il computer

Architettura di von Neumann

L'unità centrale di elaborazione (CPU)

La memoria RAM, la memoria ROM

La memoria di massa

Le periferiche di input e le periferiche di output

I sistemi posizionali

Il sistema binario

Aritmetica binaria

Il sistema esadecimale

Complemento a 2

Codifica e decodifica dell'informazione

Concetti generali sulle reti e sulle modalità di trasmissione dati

Generalità su robotica ed intelligenza artificiale

Organizzazione didattica del modulo B:

Prerequisiti
- Conoscere e saper applicare le operazioni aritmetiche di base

Obiettivi
- Conoscere le operazioni logiche fondamentali
- Saper effettuare l'analisi di un semplice circuito logico

- | |
|--|
| - Saper effettuare la sintesi di un semplice circuito logico |
|--|

Contenuti del modulo B:

Le operazioni logiche

Gli operatori logici: AND, OR, NOT, XOR, NAND, NOR

Circuiti logici

Circuiti logici a più livelli

Organizzazione didattica del modulo C:

Prerequisiti
- Conoscere le caratteristiche architettoniche di un computer

Obiettivi
- Saper realizzare il diagramma di flusso di un algoritmo
- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma
- Saper codificare un algoritmo in un linguaggio di programmazione
- Conoscere i principi della programmazione strutturata per costruire programmi ben ordinati attraverso le strutture di sequenza, selezione ed iterazione

Contenuti del modulo C:

Gli algoritmi

I diagrammi di flusso

Le strutture di controllo

La sequenza

La selezione

L'iterazione

Le basi del linguaggio C++

Dichiarazione di variabili
L'assegnazione dei valori alle variabili
Istruzioni di input e output
Istruzioni di selezione (if else switch)
Istruzioni di iterazione (for, while, do while)
Introduzione a strutture dati più complesse e alla modularità

Obiettivi minimi

- Saper elaborare una strategia risolutiva in semplici casi di problem solving
- Conoscere le caratteristiche architettoniche di un computer
- Saper rappresentare i dati con i numeri binari
- Saper effettuare conversioni tra sistema binario, sistema decimale e sistema esadecimale
- Saper rappresentare un numero in complemento a 2
- Saper effettuare operazioni di somma e sottrazione tra numeri rappresentati in complemento a 2
- Conoscere le operazioni logiche di base
- Saper realizzare il diagramma di flusso di un algoritmo che risolva semplici problemi usando le strutture di controllo di sequenza, selezione e iterazione
- Saper effettuare l'analisi di un semplice circuito logico
- Conoscere i principi della programmazione strutturata per costruire programmi attraverso le strutture di sequenza, selezione ed iterazione
- Comprendere l'importanza della modularità in programmazione
- Conoscere le caratteristiche principali delle reti e della modalità di trasmissione dei dati

Organizzazione modulare dell'insegnamento di Informatica – Classe Seconda Cambridge – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o di problem solving volte allo sviluppo del pensiero computazionale	9

A	Rappresentazione dell'informazione, trasmissione dati, memoria e dispositivi di IO	20
B	Approfondimento su porte logiche	15
C	Sicurezza ed etica	15
D	Progettazione e testing di programmi, array, sottoprogrammi	30
E	Basi di dati	10
	Totale	99

Organizzazione didattica del modulo A:

Prerequisiti
<ul style="list-style-type: none"> - Conoscere concetti base di rappresentazione dei dati e di modalità di trasmissione dati - Conoscere la struttura di un calcolatore

Obiettivi
<ul style="list-style-type: none"> - Conoscere la modalità di rappresentazione di ogni tipo di dato - Conoscere le tecniche di compressione dei dati - Essere capaci di distinguere tra trasmissione seriale, parallela, simplex, duplex, half-duplex - Capire ed essere capaci di descrivere i diversi metodi di error checking ed error detection - Capire il ruolo di un browser e di un ISP - Conoscere la struttura di un programma html e saper programmare una semplice pagina html - Conoscere i concetti di MAC address, IP address, URL e cookies - Essere capaci di descrivere i vari dispositivi di I/O e di sensori e saper illustrare caratteristiche e usi - Essere capaci di descrivere i vari dispositivi di memoria e saper illustrare caratteristiche e usi

Contenuti del modulo A:

Rappresentazione di dati, tecniche di compressione
Trasmissione seriale, parallela, simplex, duplex e half duplex
Metodi di error checking e error detection
Struttura di pagine web, concetti base di html e css, i cookie

Indirizzi MAC, IP

Tastiera, mouse, scanner, barcode readers e QR code readers

Digital cameras, touchscreen, Interactive whiteboard, microfoni

Sensori di luce, temperatura, movimento, umidità, pressione sensori ph, sensori magnetici

Display LCD e LED, proiettori LCD e DLP, stampanti, cutters e speakers

Memoria primaria, memoria secondaria e memoria off-line

Organizzazione didattica del modulo B:

Prerequisiti
- Conoscere le operazioni logiche fondamentali

Obiettivi
- Saper progettare un semplice circuito logico a partire da una descrizione verbale di un problema

Contenuti del modulo B:

Tabelle di verità

Passaggio da tabella di verità a circuito

Passaggio da circuito tabella di verità

Passaggio dalla descrizione verbale di un problema alla sua formalizzazione mediante logic statement

Passaggio da logic statement a circuito

Organizzazione didattica del modulo C:

Prerequisiti
- Conoscere la struttura di un sistema informatico

Obiettivi

- Conoscere i metodi per mantenere i dati sicuri da danneggiamenti accidentali , da attacchi informatici e da malware
- Comprendere le questioni etiche legate all'informatica
- Conoscere la differenza tra free software, freeware e shareware

Contenuti del modulo C:

Accessi non autorizzati
Hacking e cracking
Malware
Phishing e pharming
DoS e DDoS
Authentication e Biometrics
Antivirus, firewalls e proxy servers
Encryption

Organizzazione didattica del modulo D:

Prerequisiti

- Saper realizzare il diagramma di flusso di un algoritmo
- Conoscere le istruzioni principali per poter passare da un algoritmo ad un programma
- Saper codificare un algoritmo in un linguaggio di programmazione
- Conoscere i principi della programmazione strutturata per costruire programmi ben ordinati attraverso le strutture di sequenza, selezione ed iterazione

Obiettivi

- Capire e saper usare procedure e funzioni
- Essere capaci di dichiarare ed usare un array
- Saper usare top-down designs e structure diagrams
- Essere capace di effettuare il test di un programma, scegliendo in modo opportuno i test data
- Capire la necessità di effettuare validazione e verifica sugli input data
- Essere capaci di usare le trace table

- Essere capaci di trovare gli errori in un dato algoritmo
- Saper progettare ed implementare un algoritmo di ricerca lineare
- Saper progettare ed implementare un algoritmo di ordinamento bubble sort

Contenuti del modulo D:

I sottoprogrammi
Gli array
Progettazione top down
Ricerca degli errori
Trace tables
Ricerca lineare
Ordinamento. bubble sort

Organizzazione didattica del modulo E:

Prerequisiti
- Conoscere i tipi di dato

Obiettivi
- Capire cosa si intende per database - Capire i contesti in cui si utilizza un database - Conoscere la struttura di un database - Saper progettare un flat file database - Saper scegliere un tipo di dato adatto per ogni campo del database - Saper scegliere le chiavi primarie in modo appropriato - Sapere effettuare semplici query in SQL

Contenuti del modulo E:

Basi di dati

Basi di dati relazionali
Tabelle, record e campi
Flat file database
Chiave primaria
Fondamenti di SQL

Obiettivi minimi

- Conoscere la modalità di rappresentazione di ogni tipo di dato e le tecniche di compressione dei dati
- Essere capaci di distinguere tra trasmissione seriale, parallela, simplex, duplex, half-duplex
- Capire ed essere capaci di descrivere i diversi metodi di error checking ed error detection
- Conoscere la struttura di un programma html e saper programmare una semplice pagina html
- Essere capaci di descrivere i vari dispositivi di I/O e di sensori e saper illustrare caratteristiche e usi
- Essere capaci di descrivere i vari dispositivi di memoria e saper illustrare caratteristiche e usi
- Saper progettare un semplice circuito logico a partire da una descrizione verbale di un problema
- Conoscere i metodi per mantenere i dati sicuri da danneggiamenti accidentali, da attacchi informatici e da malware
- Comprendere le questioni etiche legate all'informatica
- Conoscere la differenza tra free software, freeware e shareware
- Capire e saper usare procedure e funzioni
- Essere capaci di dichiarare ed usare un array
- Essere capace di effettuare il test di un programma, scegliendo in modo opportuno i test data
- Capire la necessità di effettuare validazione e verifica sugli input data
- Essere capaci di usare le trace table
- Conoscere la struttura di un database
- Saper progettare un flat file database
- Saper scegliere un tipo di dato adatto per ogni campo del database
- Saper scegliere le chiavi primarie in modo appropriato
- Sapere effettuare semplici query in SQL

Organizzazione modulare dell'insegnamento di Informatica – Classe Terza Cambridge – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o di problem solving volte allo sviluppo del pensiero computazionale	12
A	Advanced Syllabus: Theory Fundamentals	24
B	Advanced Syllabus : Fundamental problem Solving and programming skills	24
C	Puntatori e strutture dinamiche	20
D	Programmazione lato server	19
	Totale	99

Organizzazione didattica del modulo A:

Prerequisiti
- Aver acquisito le conoscenze e competenze richieste nella prima sezione del 636089-2024-2025-syllabus

Obiettivi
- Acquisizioni degli obiettivi relativi al paper 1 elencati nella prima sezione del 636089-2024-2025-syllabus

Contenuti del modulo A: i contenuti dettagliati sono riportati nel 636089-2024-2025-syllabus e sono quelli relativi al paper 1

Information representation
Communication and network technologies
Hardware
Logic gates and logic circuits
Processor fundamentals
Assembly language programming
Monitoring and Control Systems
System Software

Security, Privacy and data integrity
Ethics and ownership
Databases

Organizzazione didattica del modulo B:

Prerequisiti
- Aver acquisito le conoscenze e competenze richieste nel syllabus Advanced

Obiettivi
- Acquisizioni degli obiettivi relativi al paper 2 elencati nel 636089-2024-2025-syllabus

Contenuti del modulo B: i contenuti dettagliati sono riportati nel 636089-2024-2025-syllabus e sono quelli relativi al paper 2

Algorithm Design and Problem Solving
Data Types and Structures
Programming and Data Representation
Software Development

Organizzazione didattica del modulo C:

Prerequisiti
- Conoscere la rappresentazione della memoria di un calcolatore
- Conoscenze gli elementi di base del linguaggio C++

Obiettivi
- Conoscere le caratteristiche di struct, puntatori, liste, pile, code
- Saper implementare in C++ le strutture dinamiche dei dati
- Saper creare una lista
- Saper visualizzare una lista

- Saper inserire un nuovo nodo in una lista in testa, in coda ed in una posizione specifica
- Saper cancellare un nodo da una lista
- Saper creare una pila
- Saper visualizzare una pila
- Saper inserire un nuovo nodo in una pila
- Saper estrarre un nodo da una pila
- Saper eliminare una pila
- Saper creare una coda
- Saper visualizzare una coda
- Saper inserire un nuovo nodo in una coda
- Saper estrarre un nodo da una coda
- Saper eliminare una coda

Contenuti del modulo C:

I record

I puntatori

Le liste: creazione di una lista, visualizzazione di una lista, inserimento di nuovi nodi in una lista in testa, in coda ed in una posizione specifica, cancellazione di un nodo da una lista

Le pile: creazione di una pila, visualizzazione di una pila, inserimento di un nodo in una pila, estrazione di un nodo da una pila, eliminazione di una pila

Le code: creazione di una coda, visualizzazione di una coda, inserimento di un nodo in una coda, estrazione di un nodo da una coda, eliminazione di una coda

Organizzazione didattica del modulo D:

Prerequisiti
- Conoscere i concetti fondamentali lato client e lato server

Obiettivi
- Saper collegare i diversi livelli di un'applicazione
- Saper individuare il livello in cui implementare una determinata funzionalità
- Saper realizzare semplici applicazioni in rete

Contenuti del modulo D:

Le componenti fondamentali di un'applicazione internet
Pagine web statiche e pagine web dinamiche
Comunicazione client-server
Elaborazione di dati prelevati da form
Memorizzazione in una base dati
Realizzazione pagine contenenti dati letti da una base dati
Implementazione di semplici applicazioni end-to-end

Obiettivi minimi

- Saper effettuare rappresentazioni in BCD e complemento a 2
- Saper descrivere caratteristiche e differenze di rappresentazioni bitmap e vector graphic
- Saper descrivere le modalità di rappresentazione del suono (campionamento e quantizzazione), sapendo effettuare valutazioni qualitative in base ai vari parametri
- Saper descrivere ed effettuare compressioni RLE e con codice di Huffman
- Saper descrivere topologie di rete, dispositivi di rete e modalità di trasmissione
- Saper descrivere i principali dispositivi hardware interni al computer e le principali periferiche
- Saper effettuare analisi e sintesi di circuiti logici
- Saper descrivere l'architettura di una CPU
- Saper comprendere un programma assembly e saper effettuare un dry run
- Conoscere le modalità di indirizzamento assembly
- Saper descrivere le varie funzionalità di un sistema operativo
- Saper elencare e illustrare i principali aspetti relativi alla sicurezza e all'etica
- Saper progettare un semplice database a partire dal modello E-R
- Conoscere le basi del linguaggio SQL
- Saper effettuare operazioni di input e output da file
- Conoscere le caratteristiche di struct, puntatori, liste, pile, code
- Saper implementare in C++ le strutture dinamiche dei dati
- Saper effettuare operazioni di base sulle strutture dinamiche
- Conoscere i principali algoritmi di ordinamento
- Saper scrivere semplici programmi lato server per il prelevamento dei dati provenienti dal client

- Saper scrivere semplici programmi lato server per la memorizzazione dei dati in una tabella
- Saper scrivere semplici programmi lato server per il recupero dei dati da una tabella

Organizzazione modulare dell'insegnamento di Informatica – Classe Quarta Cambridge – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o di problem solving volte allo sviluppo del pensiero computazionale Attività di recupero e approfondimento relative agli argomenti dell'esame Cambridge AS e alle rappresentazioni floating point, RPN e BNF	20
A	Programmazione ad oggetti: il linguaggio Java	15
B	Elementi di algebra lineare	10
C	Fondamenti di elettronica digitale: algebra booleana e circuiti logici	10
D	Riflessioni sul ruolo della tecnologia nella storia e nella società (educazione civica)	11
	Totale	66

Organizzazione didattica del modulo A:

Prerequisiti
<ul style="list-style-type: none"> - Conoscere il concetto di algoritmo - Conoscere il concetto di linguaggio di programmazione - Conoscere la struttura di un computer - Conoscere le fasi di un processo logico/tecnologico per giungere alla risoluzione di un problema in ottica informatica - Conoscere le metodologie per la stesura di un algoritmo - Conoscere le caratteristiche fondamentali delle istruzioni che compongono un algoritmo - Saper rappresentare algoritmi mediante diagrammi di flusso

Obiettivi

- | |
|---|
| <ul style="list-style-type: none">- Saper scrivere i programmi utilizzando in modo corretto la sintassi del linguaggio Java- Riconoscere le diverse fasi del lavoro di programmazione per codificare e validare gli algoritmi- Saper definire classi con attributi e metodi- Saper creare e utilizzare oggetti- Applicare i principi della programmazione ad oggetti utilizzando il linguaggio Java |
|---|

Contenuti del modulo A:

Caratteristiche generali di Java
L'ambiente di programmazione
La struttura dei programmi
Identificatori, parole chiave, variabili e costanti
La gestione dell'input/output
Le strutture di controllo: sequenza, selezione, ripetizione
La struttura di dati array
Orientamento agli oggetti
Gli oggetti e le classi
Dichiarazione e utilizzo di una classe, degli attributi e dei metodi
Creazione e utilizzo degli oggetti
Ereditarietà
Polimorfismo

Organizzazione didattica del modulo B:

Prerequisiti

- | |
|--|
| <ul style="list-style-type: none">- Conoscere il concetto di array |
|--|

Obiettivi

- | |
|--|
| <ul style="list-style-type: none">- Saper effettuare operazioni base con le matrici- Saper calcolare il determinante di una matrice- Saper risolvere un sistema lineare con i metodi di Gauss e di Gauss Jordan- Saper implementare qualche semplice programma di applicazione dei concetti base di algebra lineare |
|--|

Contenuti del modulo B:

Somma e moltiplicazioni tra matrici
Moltiplicazione per uno scalare
Matrici simmetriche, triangolari, diagonali, identità
Calcolo di un determinante
Calcolo di una matrice inversa
Metodo di Gauss
Metodo di Gauss Jordan
Teorema di Rouché Capelli
Programmi Python di implementazione dei concetti appresi

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Saper descrivere le caratteristiche di un sistema di numerazione binario- Saper convertire un numero binario in decimale- Saper convertire un numero decimale in binario

Obiettivi
<ul style="list-style-type: none">- Saper esprimere il motivo per cui si ricorre all'elettronica digitale- Saper illustrare i postulati dell'algebra di Boole- Sapere spiegare cosa si intende per rete combinatoria e per rete sequenziale- Conoscere le porte logiche fondamentali e le loro tabelle di verità- Saper definire il concetto di operatore universale- Saper effettuare la minimizzazione funzionale di una rete combinatoria- Saper illustrare il funzionamento di un half adder e di un full adder- Saper illustrare il funzionamento di un latch S-R e di un flip-flop J-K

- Saper definire e saper sintetizzare un semplice automa di Mealy o di Moore
--

Contenuti del modulo D:

Algebra di Boole
Porte logiche fondamentali
Operatori universali
Mappe di Karnaugh
Half adder
Full Adder
Latch S-R
Flip flop J-K
Automati di Mealy e di Moore

Organizzazione didattica del modulo D:

Obiettivi
- Conoscere aspetti storici relativi alla tecnologia
- Saper effettuare riflessioni critiche sulle opportunità e i rischi legati alla tecnologia

Contenuti del modulo E:

La tecnologia
Opportunità legate all'uso delle nuove tecnologie
Rischi legati all'uso delle nuove tecnologie

Per quanto riguarda la valutazione relativa ad Educazione Civica il docente potrà adottare o la griglia di istituto o una propria griglia di valutazione

Obiettivi minimi

- Utilizzare le funzioni per estrarre informazioni da una base di dati
- Saper creare un semplice database

- Saper scrivere i programmi utilizzando in modo corretto la sintassi del linguaggio Java
- Saper definire classi con attributi e metodi
- Saper creare e utilizzare oggetti
- Applicare i principi della programmazione ad oggetti utilizzando il linguaggio Java
- Saper progettare un circuito logico a partire dalla tabella di verità secondo il modello di Mealy e secondo il modello di Moore
- Conoscere le caratteristiche fondamentali della programmazione dichiarativa e in particolare di Prolog
- Saper scrivere e comprendere istruzioni in diagrammi sintattici e BNF
- Saper convertire espressioni da RPN e in RPN
- Saper distinguere i vari tipi di matrice
- Saper calcolare un determinante
- Saper calcolare l'inversa di una matrice
- Saper risolvere un sistema lineare con gli algoritmi di algebra lineare

Organizzazione modulare dell'insegnamento di Informatica – Classe Quinta Cambridge – Opzione Scienze Applicate

Elenco dei moduli in cui è stata suddivisa la disciplina:

Modulo	Titolo del modulo	Ore
0	Attività di recupero e/o di problem solving volte allo sviluppo del pensiero computazionale	10
A	Realizzazione di applicazioni a carattere scientifico	18
B	Complessità computazionale	12
C	Crittografia	12
D	Tecnologie emergenti	14
	Totale	66

Uno o più argomenti potranno essere svolti in modalità CLIL

Organizzazione didattica del modulo 0:

Obiettivi
<ul style="list-style-type: none">- Conoscere i concetti fondamentali dell'aritmetica modulare- Conoscere i concetti fondamentali dell'analisi numerica- Conoscere i concetti fondamentali dell'analisi combinatoria

Contenuti del modulo 0:

Concetti di base relativi all'aritmetica modulare
Concetti di base relativi all'analisi numerica
Concetti di base relativi all'analisi combinatoria
Concetti di base relativi al calcolo matriciale
Test di logica

Organizzazione didattica del modulo A:

Prerequisiti
<ul style="list-style-type: none">- Conoscere i concetti fondamentali relativi all'analisi numerica e all'analisi combinatoria

Obiettivi
<ul style="list-style-type: none">- Saper creare applicazioni per la risoluzione di problemi di carattere scientifico, in particolare relativi a problemi di analisi numerica e di analisi combinatoria- Saper illustrare e presentare con un linguaggio tecnicamente appropriato le applicazioni realizzate

Contenuti del modulo A:

Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo di bisezione
Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo di Newton
Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo delle secanti
Realizzazione di un'applicazione relativa al calcolo delle radici con il metodo della regula falsi
Realizzazione di un'applicazione relativa al calcolo numerico degli integrali definiti col metodo dei rettangoli
Realizzazione di un'applicazione relativa al calcolo numerico degli integrali definiti col metodo dei trapezi
Eventuale realizzazione di ulteriori applicazioni per la risoluzione di problemi a carattere scientifico

Organizzazione didattica del modulo B:

Prerequisiti
<ul style="list-style-type: none">- Saper realizzare il diagramma di flusso di un algoritmo che risolva semplici problemi usando le strutture di controllo di sequenza, selezione e iterazione- Conoscere le caratteristiche e l'andamento di una funzione polinomiale e di una funzione esponenziale
Obiettivi
<ul style="list-style-type: none">- Saper dare una definizione di complessità di un algoritmo- Saper calcolare la complessità di semplici algoritmi- Saper confrontare in termini di complessità semplici algoritmi equivalenti- Conoscere le varie notazioni asintotiche- Saper classificare i problemi come P, NP e NP--completi

Contenuti del modulo B:

Complessità di un algoritmo
La macchina di Turing
Confronto in termini di complessità tra algoritmi equivalenti
La notazione asintotica O
La notazione asintotica Ω
La notazione asintotica Θ
Classificazione dei problemi
Le classi P e NP
Problemi NP-completi

Organizzazione didattica del modulo C:

Prerequisiti
<ul style="list-style-type: none">- Acquisire i termini tecnici relativi alla crittografia- Saper applicare alcuni algoritmi elementari a chiave simmetrica per criptare un plaintext- Saper applicare alcuni algoritmi elementari a chiave simmetrica per decriptare un ciphertext- Conoscere le motivazioni che hanno reso indispensabile l'introduzione di algoritmi a chiave asimmetrica- Saper descrivere i meccanismi su cui si basano gli algoritmi a chiave asimmetrica
Obiettivi
<ul style="list-style-type: none">- Saper effettuare codifiche e decodifiche mediante cifrario di Cesare, ROT13, cifrario di Vigenere, cifrario di Bellaso- Conoscere le differenze tra un cifrario monoalfabetico ed un cifrario polialfabetico- Conoscere caratteristiche e differenze tra algoritmi a chiave simmetrica e algoritmi a chiave asimmetrica- Saper illustrare i meccanismi che regolano lo scambio di chiavi Diffie-Hellman- Saper descrivere il funzionamento dell'algoritmo a chiave asimmetrica RSA

Contenuti del modulo C:

Aspetti storici relativi alla crittografia
Cifrari monoalfabetici e polialfabetici
Cifrario di Cesare, ROT13
Cifrario di Vigenere
Cifrario di Bellaso
Algoritmi a chiave simmetrica e asimmetrica
Enigma e la sua decifrazione
Lo scambio di chiavi Diffie – Hellman
L'algoritmo a chiave asimmetrica RSA

Organizzazione didattica del modulo D:

Prerequisiti
- Conoscere i concetti fondamentali relativi alla comunicazione in rete

Obiettivi
- Saper descrivere la struttura di una rete neurale - Saper descrivere gli elementi fondamentali relativi alla progettazione di una rete neurale - Conoscere gli obiettivi della cyber security - Conoscere i principali security threats

Contenuti del modulo D:

Machine Learning
Reti neurali
Progettazione di un perceptrone
Obiettivi della cyber security
Threat actor groups
Tipi di cyber attacks

Obiettivi minimi

- Saper illustrare e presentare con un linguaggio tecnicamente appropriato applicazioni per la risoluzione di problemi di carattere scientifico, in particolare relativi a problemi di analisi numerica e di analisi combinatoria
- Saper dare una definizione di complessità di un algoritmo
- Conoscere le varie notazioni asintotiche
- Saper illustrare cosa si intende per problemi P, NP e NP—completi
- Saper illustrare i meccanismi che regolano lo scambio di chiavi Diffie-Hellman
- Saper descrivere il funzionamento dell'algoritmo a chiave asimmetrica RSA
- Saper descrivere la struttura di una rete neurale
- Saper descrivere gli elementi fondamentali relativi alla progettazione di una rete neurale
- Conoscere gli obiettivi della cyber security
- Conoscere i principali security threats