

**CLASSE: 1 e 2 Liceo Scienze Applicate**

**MATERIA: Informatica**

**COMPETENZE CHIAVE DI CITTADINANZA**  
**relative all'intero curricolo della disciplina**

1. competenza alfabetica funzionale (saper comunicare in forma scritta e orale)
2. competenza digitale (conoscere e utilizzare programmi di videoscrittura, fogli di calcolo e presentazioni)
3. competenza personale, sociale e capacità di imparare ad imparare (applicare strategie efficaci di apprendimento; lavorare in gruppo in maniera costruttiva)
4. competenza imprenditoriale (risolvere problemi)

COMPETENZE	ABILITA'	CONOSCENZE
<ul style="list-style-type: none"><li>● Riconoscere la struttura di un sistema di elaborazione</li><li>● Distinguere le tipologie dei computer in base alle attività e alle caratteristiche</li><li>● Riconoscere il ruolo dei componenti di un sistema di elaborazione</li><li>● Convertire numeri e codici rappresentati secondo sistemi di numerazione diversi</li></ul>	<ul style="list-style-type: none"><li>● Identificare i componenti hardware di un computer</li><li>● Utilizzare in modo appropriato la terminologia tecnica</li><li>● Confrontare le caratteristiche tecniche principali dei singoli componenti</li><li>● Codificare e decodificare numeri e codici</li></ul>	<p><b>Introduzione all'architettura del computer</b></p> <ul style="list-style-type: none"><li>● L'architettura ed i componenti fondamentali di un computer</li><li>● Le memorie: tipologie e capacità</li><li>● Sistema di numerazione decimale binario ottale esadecimale.</li><li>● Il software</li></ul>
<ul style="list-style-type: none"><li>● Utilizzare le procedure necessarie per gestire le impostazioni</li></ul>	<ul style="list-style-type: none"><li>● Utilizzare in modo appropriato la terminologia tecnica</li></ul>	<p><b>Il funzionamento del Sistema Operativo</b></p> <ul style="list-style-type: none"><li>● Riconoscere il ruolo dei sistemi operativi e quali sono i tipi più diffusi</li></ul>

<ul style="list-style-type: none"> <li>• dello schermo e del desktop</li> <li>• Utilizzare le tecniche di drag and drop per gestire i file e le cartelle</li> <li>• Essere in grado di estrarre le caratteristiche del computer in uso</li> <li>• Gestire i file system di windows e di Linux Ubuntu</li> </ul>	<ul style="list-style-type: none"> <li>• Riconoscere le caratteristiche principali del sistema operativo</li> </ul>	<ul style="list-style-type: none"> <li>• Conoscere le caratteristiche principali del desktop di windows e di linux</li> <li>• Saper distinguere i tipi di file in base all'estensione</li> <li>• Riconoscere il significato e la struttura delle directory</li> <li>• Identificare i principali elementi dell'interfaccia grafica di windows e linux ubuntu.</li> <li>• Identificare il significato dei caratteri jolly</li> </ul>
<ul style="list-style-type: none"> <li>• Conoscere e saper utilizzare Microsoft word.</li> <li>• Conoscere e saper utilizzare PowerPoint</li> <li>• Conoscere e saper utilizzare excel, saper applicare formule e funzioni corrette in base al contesto</li> </ul>	<ul style="list-style-type: none"> <li>• Realizzare documenti word e lettere circolari</li> <li>• Realizzare presentazioni multimediali con PowerPoint</li> <li>• Applicare le funzioni condizionali ai fogli di lavoro in situazioni complesse.</li> </ul>	<p><b>Elaborazione di documenti digitali</b></p> <ul style="list-style-type: none"> <li>• Elementi fondamentali del documento Word / Google Documenti</li> <li>• Elementi fondamentali di una presentazione PowerPoint/Google Slides</li> <li>• Riconoscere le caratteristiche del foglio di calcolo di Excel o dei fogli di Google (Google sheets)</li> <li>• Orientarsi nell'ambiente di lavoro Excel/Fogli di Google inserendo formule e funzioni per la produttività personale.</li> </ul>
<ul style="list-style-type: none"> <li>• Descrivere la soluzione di semplici problemi mediante algoritmi.</li> <li>• capire il concetto di programma e codice sorgente</li> </ul>	<ul style="list-style-type: none"> <li>• Utilizzare i diagrammi di flusso per codificare gli algoritmi</li> <li>• Saper scrivere compilare ed eseguire un primo semplice programma.</li> </ul>	<p><b>Principi base di programmazione</b></p> <ul style="list-style-type: none"> <li>• Concetto di Linguaggio di programmazione</li> <li>• Concetto di algoritmo</li> <li>• Conoscere la simbologia dei diagrammi di flusso.</li> </ul>
<ul style="list-style-type: none"> <li>• Editare testare e collaudare un programma in Python</li> <li>• Effettuare l'input dei dati</li> <li>• Effettuare l'output dei dati</li> <li>• Scrivere programmi con</li> </ul>	<ul style="list-style-type: none"> <li>• Installare e configurare l'ambiente di sviluppo (Pycharm)</li> <li>• Disporre l'output sullo schermo</li> <li>• Utilizzare le variabili nei programmi</li> <li>• Commentare il codice del programma</li> <li>• Utilizzare variabili intere, reali e bool</li> </ul>	<p><b>Programmare in Python</b></p> <ul style="list-style-type: none"> <li>• Conoscere i diversi formati di un programma</li> <li>• Descrivere le diverse fasi di sviluppo di un programma</li> <li>• Comprendere il concetto di variabile</li> <li>• Conoscere le istruzioni di comunicazione con l'utente</li> </ul>

più istruzioni in sequenza.	<ul style="list-style-type: none"> <li>Utilizzare gli operatori / e % sui numeri interi</li> </ul>	<ul style="list-style-type: none"> <li>Comprendere l'importanza del commento del codice</li> </ul>
<ul style="list-style-type: none"> <li>Riconoscere quando utilizzare selezioni annidate</li> <li>Codificare la selezione semplice e doppia</li> <li>Effettuare l'annidamento delle istruzioni</li> <li>Utilizzare le variabili di tipo bool</li> <li>Utilizzare gli operatori logici</li> </ul>	<ul style="list-style-type: none"> <li>Scrivere codice con istruzioni condizionali</li> <li>Scrivere codice con blocchi di istruzioni annidate</li> <li>Combinare più selezioni con condizioni logiche composte</li> <li>Scrivere codice utilizzando l'istruzione di selezione multipla</li> </ul>	<p><b>Blocco di Selezione (if then else elif)</b></p> <ul style="list-style-type: none"> <li>Conoscere l'istruzione di selezione semplice e doppia</li> <li>Conoscere le variabili di tipo bool e gli operatori logici</li> <li>Comprendere il concetto di annidamento</li> </ul>
<ul style="list-style-type: none"> <li>Saper scegliere il tipo di iterazione adeguato alle diverse situazioni.</li> <li>Codificare l'iterazione finita ed indefinita.</li> <li>Codificare programmi con cicli annidati.</li> </ul>	<ul style="list-style-type: none"> <li>Scrivere programmi con selezioni e iterazioni annidate.</li> <li>Progettare programmi con cicli e cicli annidati.</li> <li>Utilizzare diverse tipologie di iterazione nello stesso programma.</li> </ul>	<p><b>Blocco di Iterazione (for,while)</b></p> <ul style="list-style-type: none"> <li>Comprendere il concetto di iterazione.</li> <li>Conoscere le diverse tipologie di iterazione</li> </ul>
<ul style="list-style-type: none"> <li>Definire una funzione in Python.</li> <li>Distinguere tra parametri formali e parametri attuali.</li> </ul>	<ul style="list-style-type: none"> <li>Scrivere programmi in Python suddividendoli in moduli tramite le funzioni.</li> <li>Utilizzare funzioni predefinite nei programmi.</li> <li>Scrivere funzioni ricorsive.</li> <li>Trasformare funzioni iterative in funzioni ricorsive.</li> </ul>	<p><b>Le Funzioni</b></p> <ul style="list-style-type: none"> <li>Comprendere il concetto di funzione nella programmazione.</li> <li>Conoscere il concetto di invocazione di una funzione e di passaggio dei parametri.</li> <li>Conoscere i Record di Attivazione ed il loro funzionamento.</li> <li>Comprendere il concetto di ricorsione e la differenza rispetto all'iterazione.</li> </ul>

<ul style="list-style-type: none"> <li>• Saper definire una stringa in Python e saperla manipolare.</li> </ul>	<ul style="list-style-type: none"> <li>• Scrivere programmi in Python che lavorano con le stringhe.</li> <li>• Utilizzare le principali operazioni e funzioni con le stringhe.</li> </ul>	<p><b>Le stringhe</b></p> <ul style="list-style-type: none"> <li>• Conoscere la struttura dati Stringa.</li> <li>• Comprendere l'utilità del tipo di dato Stringa</li> <li>• Conoscere quali operazioni sono possibili su una stringa.</li> </ul>
--	---	---

*N.B.: le conoscenze devono essere riferite alle Linee Guida per i Licei del 2010  
- DPR 89*

**CLASSE: 3 e 4 Liceo Scienze Applicate**

**MATERIA: Informatica**

### **COMPETENZE CHIAVE DI CITTADINANZA**

#### **relative all'intero curricolo della disciplina**

1. competenza alfabetica funzionale (saper comunicare in forma scritta e orale)
2. competenza digitale (conoscere e utilizzare programmi per lo sviluppo e il testing di programmi)
3. competenza personale, sociale e capacità di imparare ad imparare (applicare strategie efficaci di apprendimento; lavorare in gruppo in maniera costruttiva)
4. competenza imprenditoriale (risolvere problemi)

COMPETENZE	ABILITA'	CONOSCENZE
<ul style="list-style-type: none"><li>• Saper creare una lista in Python saperla utilizzare ed iterare.</li><li>• Comprendere la sintassi di List Comprehension e sapere quando utilizzarla.</li></ul>	<ul style="list-style-type: none"><li>• Scrivere programmi/funzioni in Python che lavorano con le Liste.</li><li>• Utilizzare le principali operazioni e funzioni con le Liste.</li><li>• Utilizzare la sintassi di list comprehension per costruire Liste.</li></ul>	<p><b>Le liste</b></p> <ul style="list-style-type: none"><li>• Conoscere il tipo di dato Lista.</li><li>• Comprendere l'utilità del tipo di dato Lista.</li><li>• Conoscere quali operazioni sono possibili su una lista e in che contesti viene utilizzata.</li><li>• Conoscere la sintassi di List Comprehension.</li></ul>
<ul style="list-style-type: none"><li>• Saper quando è necessario l'utilizzo di una tupla in un programma Python.</li></ul>	<ul style="list-style-type: none"><li>• Fare “unpacking” di tuple.</li></ul>	<p><b> Tuple</b></p> <ul style="list-style-type: none"><li>• Conoscere il tipo di dato Tupla.</li></ul>
<ul style="list-style-type: none"><li>• Saper creare un dizionario in Python, saperlo utilizzare ed iterare.</li><li>• Comprendere la sintassi di Dictionary Comprehension e</li></ul>	<ul style="list-style-type: none"><li>• Scrivere programmi/funzioni in Python che lavorano con i Dizionari.</li><li>• Utilizzare le principali operazioni e funzioni con i dizionari.</li><li>• Utilizzare la sintassi di Dictionary</li></ul>	<p><b>Dizionari</b></p> <ul style="list-style-type: none"><li>• Conoscere il tipo di dato Dizionario.</li><li>• Comprendere l'utilità del tipo di dato Dizionario.</li><li>• Conoscere quali operazioni sono possibili su un Dizionario e in che contesti viene utilizzato.</li></ul>

sapere quando utilizzarla.	Comprehension per costruire Dizionari.	<ul style="list-style-type: none"> <li>Conoscere la sintassi di Dictionary Comprehension.</li> </ul>
<ul style="list-style-type: none"> <li>Saper creare un set in Python, saperlo utilizzare ed iterare.</li> </ul>	<ul style="list-style-type: none"> <li>Scrivere programmi/funzioni in Python che lavorano con i Set.</li> <li>Utilizzare le principali operazioni e funzioni sui Set.</li> </ul>	<b>Set</b> <ul style="list-style-type: none"> <li>Conoscere il tipo di dato Set.</li> <li>Comprendere l'utilità del tipo di dato Set.</li> <li>Conoscere quali operazioni sono possibili su un Dizionario e in che contesti viene utilizzato.</li> </ul>
<ul style="list-style-type: none"> <li>Saper aprire un file in lettura ed in scrittura mediante la primitiva open</li> <li>Saper usare le funzioni dell'oggetto file in Python.</li> </ul>	<ul style="list-style-type: none"> <li>scrivere programmi che facciano input e output da e su file.</li> </ul>	<b>Lettura/Scrittura su file (Python)</b> <ul style="list-style-type: none"> <li>Conoscere le diverse modalità di apertura di un file e la loro differenza.</li> </ul>
<ul style="list-style-type: none"> <li>Definire in Python una classe con attributi e metodi.</li> <li>Classificare classi e relazioni tra esse.</li> <li>Riconoscere e progettare la gerarchia tra classi.</li> </ul>	<ul style="list-style-type: none"> <li>Usare la progettazione orientata agli oggetti per i programmi complessi.</li> <li>Applicare il concetto di astrazione per modellare le classi.</li> <li>Individuare la specializzazione e la generalizzazione di una classe.</li> <li>Applicare i concetti di ereditarietà e di polimorfismo nello sviluppo di programmi.</li> </ul>	<b>Programmazione ad Oggetti</b> <ul style="list-style-type: none"> <li>Conoscere gli elementi teorici del paradigma di programmazione ad oggetti.</li> <li>Comprendere il concetto di astrazione nell'ambito della programmazione ad oggetti.</li> <li>Comprendere il concetto di ereditarietà e di polimorfismo.</li> </ul>

*N.B.: le conoscenze devono essere riferite alle Linee Guida per i Licei del 2010  
- DPR 89*

CLASSE: **5 Liceo Scienze Applicate**

MATERIA: **Informatica**

## **COMPETENZE CHIAVE DI CITTADINANZA<sup>1</sup>**

### **relative all'intero curricolo della disciplina**

1. competenza alfabetica funzionale (saper comunicare in forma scritta e orale)
2. competenza digitale (conoscere e utilizzare programmi di videoscrittura, fogli di calcolo e presentazioni)
3. competenza personale, sociale e capacità di imparare ad imparare (applicare strategie efficaci di apprendimento; lavorare in gruppo in maniera costruttiva)
4. competenza imprenditoriale (risolvere problemi)

<b>COMPETENZE</b>	<b>ABILITA'</b>	<b>CONOSCENZE</b>
<ul style="list-style-type: none"><li>• Utilizzare lo schema dei dati E-R</li><li>• Utilizzare il modello logico dei dati</li><li>• Utilizzare gli operatori relazionali</li><li>• Rispettare le regole di integrità</li><li>• Effettuare ricerche nelle tabelle</li><li>• Interrogare il database mediante query</li><li>• Definire un database con il DBMS MySQL</li><li>• Realizzare relazioni tra tabelle</li></ul>	<ul style="list-style-type: none"><li>• Applicare le gerarchie di generalizzazione</li><li>• Utilizzare le potenzialità di una base di dati relazionale</li><li>• Applicare le regole di normalizzazione</li><li>• Progettare basi di dati relazionali</li></ul>	<p><b>Database</b></p> <ul style="list-style-type: none"><li>• Comprendere le necessità dei database</li><li>• Conoscere i vantaggi di un DBMS</li><li>• Individuare le entità e le relazioni all'interno di una situazione complessa</li><li>• Acquisire la conoscenza degli aspetti funzionali e organizzativi di una base di dati</li><li>• Conoscere il concetto di dipendenza funzionale</li><li>• Conoscere le motivazioni alla base della normalizzazione</li><li>• Conoscere il significato di database</li><li>• Riconoscere il ruolo dei DBMS</li><li>• conoscere lla struttura di tabelle record e campi</li><li>• conoscere il significato di relazione e di chiave primaria</li></ul>

<ul style="list-style-type: none"> <li>• realizzare query personalizzate</li> </ul>		
<ul style="list-style-type: none"> <li>• Convertire in base 10 un numero relativo rappresentato in complemento a due.</li> <li>• Convertire in base 10 un numero reale rappresentato in virgola fissa ed in virgola mobile</li> <li>• Saper distinguere errore relativo ed errore assoluto, saper arrotondare e troncare.</li> <li>• Codificare l'algoritmo babilonese e di newton per il calcolo della radice quadrata.</li> <li>• Codificare il metodo di bisezione per trovare gli zeri di una funzione.</li> </ul>	<ul style="list-style-type: none"> <li>• Saper rappresentare un numero relativo in complemento a due</li> <li>• Saper rappresentare un numero reale in virgola mobile.</li> <li>• Calcolare errore assoluto ed errore relativo.</li> <li>• Saper implementare il calcolo numerico della radice quadrata di un numero</li> <li>• Saper implementare il metodo di bisezione.</li> </ul>	<p><b>Algoritmi del calcolo numerico</b></p> <ul style="list-style-type: none"> <li>• Comprendere le diverse rappresentazioni dei numeri nel calcolatore.</li> <li>• Comprendere il concetto di errore assoluto ed errore relativo dovuto all'approssimazione.</li> <li>• Calcolo approssimato della radice quadrata.</li> <li>• Calcolo approssimato della radice di un equazione con il metodo di bisezione</li> </ul>

<ul style="list-style-type: none"> <li>• Classificare le reti in base alla topologia</li> <li>• Individuare i diversi dispositivi di rete</li> <li>• Saper classificare le reti in base ai mezzi trasmissivi</li> <li>• Classificare le tecniche di trasferimento dell'informazione.</li> </ul>	<ul style="list-style-type: none"> <li>• Riconoscere le funzioni in relazione ai diversi livelli protocollari</li> <li>• Confrontare il modello ISO/OSI con il modello TCP/IP</li> <li>• Delineare i compiti del modello ISO/OSI e TCP/IP</li> <li>• Scomporre una rete in sottoreti</li> </ul>	<p><b>Le Reti</b></p> <ul style="list-style-type: none"> <li>• Conoscere gli elementi fondamentali di una rete</li> <li>• Conoscere le topologie di rete</li> <li>• Acquisire il concetto di protocollo</li> <li>• Conoscere come si è sviluppata internet ed il protocollo TCP/IP</li> <li>• Il confronto tra i livelli ISO/OSI e TCP/IP</li> <li>• I 4 strati del modello TCP/IP e le loro funzioni</li> <li>• La struttura degli indirizzi IP</li> <li>• Le classi degli indirizzi IP</li> <li>• Differenze tra indirizzamento pubblico e privato.</li> </ul>
---	---	--

<sup>1</sup> – <sup>2</sup> *vedi file dei materiali*

*N.B.: le conoscenze devono essere riferite alle Linee Guida per i Licei del 2010*  
*- DPR 89*