

**LICEO SCIENTIFICO A. GRAMSCI DI IVREA**

**ANNO SCOLASTICO 2016/2017**

**PIANO DI LAVORO GENERALE  
INFORMATICA  
LICEO SCIENZE APPLICATE**

**SECONDO BIENNIO**

**CLASSE 4E, 4F, 4G**

## FINALITA'

L'insegnamento di INFORMATICA nel secondo biennio si propone di:

- **potenziare** l'uso degli strumenti multimediali a supporto dello studio e della ricerca
- **far comprendere** il ruolo della tecnologia come mediazione fra scienza e vita quotidiana
- **abituare** alla precisione di linguaggio
- **sostenere e favorire** il processo di astrazione stimolando la capacità di riflessione razionale
- **potenziare** le capacità analitiche e sintetiche
- **creare** la consapevolezza della necessità dei processi di formalizzazione
- **impostare** un corretto approccio metodologico nell'analisi delle situazioni di incertezza.

## COMPETENZE

Nel corso del secondo biennio l'allievo deve acquisire **competenze** atte a

- **comprendere** la valenza metodologica dell'informatica nella formalizzazione e modellizzazione dei processi complessi e nell'individuazione di procedimenti risolutivi
- **individuare** la funzione dell'informatica nello sviluppo scientifico.
- **essere in grado** di utilizzare criticamente i principali strumenti informatici e telematici nelle attività di studio e di approfondimento
- **utilizzare** gli strumenti informatici in relazione all'analisi dei dati e alla modellizzazione di specifici problemi scientifici
- **utilizzare** tali strumenti per la soluzione di problemi significativi in generale, ma in particolare connessi allo studio delle 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
- **affrontare** con approccio razionale situazioni in vari ambiti disciplinari
- **applicare** in modo consapevole le diverse tecniche operative
- **applicare** con autonomia crescente il metodo ipotetico-deduttivo
- **esprimersi** con rigore logico e linguistico.

## CAPACITA'

Lo studente deve inoltre maturare in sé la **capacità** di:

- **riflessione razionale su temi già affrontati in modo intuitivo**
- **ragionamento coerente ed argomentato**
- **induzione e deduzione.**

## OBIETTIVI SPECIFICI E CONTENUTI

Le finalità sopra indicate vengono qui specificate in relazione ad ognuna delle aree tematiche attorno a cui si organizzano e si sviluppano i **contenuti**, che sono presentati nel **programma annuale**.

Per ogni tema sono indicati gli obiettivi e i saperi essenziali afferenti la specifica classe.

## **CLASSE QUARTA**

### **Area tematica: Algoritmi e linguaggi di programmazione (AL)**

#### **Obiettivi**

Utilizzo del linguaggio C:

- Saper utilizzare dati di tipo semplice, strutture di controllo, dati strutturati e funzioni nel linguaggio C
- Saper tradurre algoritmi in codice
- Risolvere autonomamente problemi che richiedono l'organizzazione dei dati in strutture statiche
- Risolvere autonomamente problemi che richiedono l'organizzazione della soluzione in funzioni

Programmazione orientata agli oggetti:

- Comprendere la differenza tra la tecnica di programmazione top-down e bottom-up
- Comprendere i concetti alla base della programmazione orientata agli oggetti: classi, attributi, metodi, information hiding e incapsulamento
- Saper definire, implementare e far interagire tra loro oggetti
- Progettare la soluzione di un problema in forma grafica (UML)
- Comprendere le differenze tra il linguaggio C e il linguaggio C++
- Saper tradurre la soluzione di un problema in codice utilizzando il linguaggio C++
- Comprendere i concetti avanzati della programmazione orientata agli oggetti: ereditarietà e polimorfismo
- Progettare la soluzione di un problema che richiede l'uso di ereditarietà e polimorfismo (UML)

Utilizzo di AppInventor:

- Saper utilizzare l'ambiente di programmazione grafica AppInventor per costruire applicazioni per dispositivi mobili
- Saper utilizzare l'ambiente di programmazione AppInventor per costruire semplici applicazioni di elaborazione numerica e semplici giochi.

Utilizzo di Microsoft Visual Studio C#

- Saper utilizzare l'ambiente di programmazione C# per costruire semplici applicazioni di elaborazione numerica.

### **Area tematica: Basi di dati (BD)**

#### **Obiettivi**

- Comprendere la necessità dei DB
- Comprendere la funzione svolta e le caratteristiche di un DBMS
- Progettare la soluzione di una situazione reale utilizzando il modello concettuale E/R
- Tradurre lo schema concettuale in modello relazionale
- Applicare gli operatori dell'algebra relazione per interrogare la base di dati
- Utilizzare il linguaggio SQL per creare un database, costruire tabelle, popolare tabelle, costruire query semplici e annidate

## CONTENUTI

### Area tematica: Algoritmi e linguaggi di programmazione (AL)

Utilizzo del linguaggio C:

- Dichiarazione, uso, lettura e visualizzazione di variabili di tipo numerico (intero, reale)
- Traduzione delle strutture di controllo (selezione, ciclo a condizione iniziale, finale, conteggio)
- Dichiarazione, lettura e visualizzazione di variabili di tipo vettoriale
- Uso di vettori per svolgere operazioni semplici, ricerche e ordinamenti
- Organizzazione della soluzione di un problema in funzioni
- Definizione, implementazione e uso di una funzione
- Risoluzione di problemi che richiedono dati di tipo semplice o strutturato e funzioni

Programmazione orientata agli oggetti:

- Confronto tra la tecnica di programmazione top-down e bottom-up
- I concetti alla base della programmazione orientata agli oggetti: classi, attributi, metodi, information hiding e incapsulamento
- Definizione e implementazione di oggetti
- Progettazione di soluzione di problemi in forma grafica (UML)
- Le differenze tra il linguaggio C e il linguaggio C++
- Traduzione della soluzione di un problema in codice utilizzando il linguaggio C++
- I concetti avanzati della programmazione orientata agli oggetti: ereditarietà e polimorfismo
- Progettazione della soluzione di problemi che richiede l'uso di ereditarietà e polimorfismo (UML)

AppInventor:

- Utilizzo dell'ambiente di programmazione grafica AppInventor per costruire applicazioni per dispositivi mobili
- Utilizzo dell'ambiente di programmazione AppInventor per costruire semplici applicazioni di elaborazione numerica e semplici giochi

Microsoft Visual Studio C#

- Utilizzare l'ambiente di programmazione C# per costruire semplici applicazioni di elaborazione numerica.

### Area tematica: BD Basi di dati (BD)

- Sistemi informativi (dati, informazioni, risorse e processi)
- Modello dei dati
- DBMS
- La progettazione concettuale: il modello E/R
  - Entità e attributi
  - Associazioni (tipi e proprietà)
  - Vincoli di integrità
  - Chiavi: candidate, primaria e straniera
- Traduzione dello schema E/R in schema logico relazionale (mapping)

- Normalizzazione (cenni)
- Lo standard SQL
  - Istruzioni per la dichiarazione dei dati (DDL)
  - Istruzioni per manipolazione dei dati (DML)
  - Reperimento dati: SELECT
  - Le operazioni relazionali (proiezione, selezione, giunzione)
  - Query parametriche
  - Funzioni di aggregazione
  - Query annidate

## **OBIETTIVI DIDATTICI MINIMI**

### **Obiettivi minimi**

Utilizzo del linguaggio C:

- Saper utilizzare dati di tipo semplice, strutture di controllo, dati strutturati e funzioni nel linguaggio C
- Saper tradurre algoritmi in codice
- Risolvere autonomamente problemi che richiedono l'organizzazione dei dati in strutture statiche
- Risolvere autonomamente problemi che richiedono l'organizzazione della soluzione in funzioni

Programmazione orientata agli oggetti:

- Comprendere la differenza tra la tecnica di programmazione top-down e bottom-up
- Comprendere i concetti alla base della programmazione orientata agli oggetti: classi, attributi, metodi, information hiding e incapsulamento
- Saper definire, implementare e far interagire tra loro oggetti
- Progettare la soluzione di un problema in forma grafica (UML)
- Comprendere le differenze tra il linguaggio C e il linguaggio C++
- Saper tradurre la soluzione di un problema semplice in codice utilizzando il linguaggio C++

Utilizzo di AppInventor:

- Saper utilizzare l'ambiente di programmazione grafica AppInventor per costruire applicazioni per dispositivi mobili
- Saper utilizzare l'ambiente di programmazione AppInventor per costruire semplici applicazioni di elaborazione numerica e semplici giochi.

### **Area tematica: Basi di dati (BD)**

#### **Obiettivi minimi**

- Comprendere la necessità dei DB
- Comprendere la funzione svolta e le caratteristiche di un DBMS
- Progettare la soluzione di una situazione reale utilizzando il modello concettuale E/R
- Tradurre lo schema concettuale in modello relazionale
- Applicare gli operatori dell'algebra relazione per interrogare la base di dati
- Utilizzare il linguaggio SQL per costruire query semplici

## **METODOLOGIA DI LAVORO**

La metodologia di lavoro dovrà favorire la partecipazione attiva degli allievi al processo educativo e l'acquisizione di una progressiva autonomia operativa, attraverso il potenziamento delle conoscenze e delle abilità specifiche. Gli studenti dovranno essere in grado di usare le tecniche e gli strumenti informatici per la soluzione di problemi significativi in generale, ma in particolare connessi allo studio delle altre discipline. Per questa ragione le unità didattiche verranno trattate, di volta in volta, nel periodo temporale più adatto a favorire la migliore interdisciplinarietà.

Il rapporto fra teoria e pratica sarà mantenuto su di un piano paritario e i due aspetti saranno strettamente integrati.

Si cercherà inoltre, ove possibile, di preparare gli studenti secondo gli standard europei previsti da ECDL. Prendendo anche spunto dagli interessi personali degli allievi si cercherà infine di favorire l'approfondimento individuale e l'attività di ricerca su temi (anche interdisciplinari) collegati al programma affrontato.

Come strumenti si utilizzeranno la lezione frontale, la lezione dialogata e partecipata, le discussioni collettive e il lavoro di gruppo. Alla lezione frontale si farà ricorso prevalentemente nella fase di sistemazione teorica dei contenuti, mentre il lavoro individuale o di gruppo sarà utilizzato soprattutto nella fase di ricerca e di risoluzione di problemi.

L'attività di laboratorio, strettamente collegata agli argomenti trattati, consisterà nell'uso di strumenti hardware e nella realizzazione di procedure software.

## **STRUMENTI DI VERIFICA E CRITERI DI VALUTAZIONE**

Tenuti presenti gli obiettivi formativi e didattici da raggiungere ed i tempi di assimilazione di ciascuna classe, si prevede di verificare il livello raggiunto nella conoscenza degli argomenti trattati con verifiche scritte, orali e pratiche.

Le verifiche scritte avverranno attraverso vari tipi di prove: test a risposta chiusa e a risposta aperta, esercitazioni su piccoli segmenti di unità, questionari su temi di tipo teorico, anche su intere unità, scrittura di segmenti di programmi. Mediante tali prove si intendono valutare, oltre al possesso dei contenuti, anche le abilità raggiunte nell'utilizzare le strutture dati, le tecniche e gli strumenti più adeguati alla risoluzione del quesito proposto, nonché la capacità di affrontare situazioni nuove in contesti noti e di applicare, in modo sempre più autonomo, le conoscenze e le tecniche apprese alla soluzione di problemi in contesti via via più complessi.

Verranno proposte, se possibile, verifiche comuni per classi parallele.

Le prove orali, intese sia come brevi risposte dal banco sia come interrogazioni vere e proprie, saranno volte a valutare la capacità di sistemazione e rielaborazione teorica delle conoscenze, l'uso del linguaggio specifico e del formalismo informatico, la capacità di impostare e svolgere correttamente ragionamenti deduttivi e di effettuare collegamenti.

Le prove di laboratorio verranno valutate tenendo conto sia del metodo scelto dallo studente per affrontare il problema proposto, sia del grado di autonomia manifestato durante l'esecuzione del lavoro e del rispetto delle scadenze.

Il voto numerico assegnato all'allievo al termine di ogni quadrimestre sarà l'espressione sintetica di un giudizio alla cui formulazione concorrono:

- la preparazione di base e il percorso compiuto
- la conoscenza dei contenuti, sia in termini quantitativi che qualitativi
- l'esposizione scritta e orale con particolare attenzione all'uso della terminologia specifica e del formalismo proprio della disciplina
- la capacità di analisi e risoluzione dei problemi
- il possesso di strumenti operativi e di tecniche specifiche, con particolare attenzione agli strumenti hardware e software

- la capacità di effettuare sintesi e collegamenti, sfruttando adeguatamente sia le doti intuitive che quelle razionali
- l'impegno e il rispetto delle scadenze.

Per quanto riguarda la scala di valutazione si fa riferimento alla **Scheda di valutazione delle prove sommative in decimi** approvata dal Collegio Docenti e inserita nel POF adattata agli obiettivi della Disciplina e di seguito riportata.

### ATTIVITA' DI RECUPERO E DI SOSTEGNO

L'insegnante prevede di organizzare individualmente, laddove se ne verifichi la necessità, interventi di recupero in itinere.

Nel corso dell'anno scolastico potranno essere concordati e attivati, su richiesta di singoli o piccoli gruppi di studenti, sia lo Sportello DIDATTICO che lo Sportello HELP.

### TABELLA DI VALUTAZIONE

Di seguito viene presentato un prospetto contenente i criteri di valutazione a cui il dipartimento intende fare riferimento.

CONOSCENZA	VOTO
Nessuna	1-2
Assenza conoscenza basilari, gravi ed estese lacune	3-4
Frammentaria e superficiale	5
Adeguata	6
Completa	7
Completa e approfondita	8
Completa, ben coordinata e personalizzata	9-10
COMPRESIONE	
Nessuna	1-2
Gravi difficoltà su contenuti semplici	3-4
Alcune difficoltà su contenuti semplici	5
Nessuna difficoltà su contenuti semplici	6
Alcune difficoltà e imprecisioni su contenuti complessi	7
Nessuna difficoltà su contenuti complessi	8
Facilità su contenuti anche molto complessi	9-10
APPLICAZIONE	
Non riesce ad effettuare applicazioni	1-2
Non riesce ad applicare le conoscenze in contesti noti	3-4
Applica, aiutato, le conoscenze, in contesti noti	5
Sa applicare le conoscenze in contesti semplici senza commettere gravi errori	6
Sa applicare le conoscenze in ambiti semplici, senza errori	7
Sa applicare i contenuti anche in ambiti complessi con qualche imprecisione	8
Applica le conoscenze in contesti nuovi senza errori e con spunti personali	9-10
ESPOSIZIONE	
Assenza utilizzo linguaggio specifico	1-2
Gravi carenze lessicali e logico-sintattiche	3-4
Lessico con improprietà o povero, sintassi debole	5

Lessico adeguato, sintassi nei limiti della correttezza	6
Lessico più ampio e appropriato, sintassi più articolata	7
Lessico ricco, sintassi sciolta e ben articolata	8
Grande precisione e ricchezza di espressione	9-10