



ISTITUTO ISTRUZIONE SUPERIORE

“Federico II di Svevia”

Liceo Scientifico – Classico – Linguistico e Scienze Applicate

Via G. Verdi, 1 – 85025 MELFI (PZ)

Tel. 097224434/35 Cod. Min.: PZIS02700B Cod. Fisc. 85001210765

e-mail: pzis02700b@istruzione.it sito: www.liceomelfi.it

Liceo Artistico Statale “Festa Campanile” Piazza Mazzini – 85025 MELFI (PZ) Tel. 097221131



INFORMATICA

Liceo delle Scienze Applicate

Classe 2[^] AA

PIANO DI STUDIO

a.s. 2018/2019

Professoressa M. C. Navazio



Informatica

Liceo delle Scienze Applicate

	1° biennio		2° biennio		5°
	1^	2^	3^	4^	5^
Informatica	66	66	66	66	66

Premessa generale.

In base alle indicazioni nazionali concernenti gli obiettivi per il Liceo delle Scienze Applicate, l'insegnamento di informatica deve temperare diversi obiettivi: comprendere i principali fondamenti teorici delle scienze dell'informazione, acquisire padronanza del linguaggio tecnico, logico e formale della disciplina, comprendere la valenza metodologica dell'informatica nella formalizzazione e modellizzazione dei processi complessi e nell'individuazione di procedimenti risolutivi, acquisire la padronanza di strumenti dell'informatica, 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.

Questi obiettivi si riferiscono ad aspetti fortemente connessi fra di loro, che vanno quindi trattati in modo integrato. Il rapporto fra teoria e pratica va mantenuto su di un piano paritario e i due aspetti vanno strettamente integrati evitando sviluppi paralleli incompatibili con i limiti del tempo a disposizione.

L'uso di strumenti e la creazione di applicazioni deve essere accompagnata non solo da una conoscenza adeguata delle funzioni e della sintassi, ma da un sistematico collegamento con i concetti teorici ad essi sottostanti.

Il collegamento con le discipline scientifiche, ma anche con la filosofia e l'italiano, deve permettere di riflettere sui fondamenti teorici dell'informatica e delle sue connessioni con la logica, sul modo in cui l'informatica influisce sui metodi delle scienze e delle tecnologie e su come permette la nascita di nuove scienze.

È opportuno coinvolgere gli studenti degli ultimi due anni in percorsi di approfondimento anche mirati al proseguimento degli studi universitari e di formazione superiore. In questo contesto è auspicabile trovare un raccordo con altri insegnamenti, in particolare con matematica, fisica e scienze, e sinergie con il territorio, aprendo collaborazioni con università, enti di ricerca, musei della scienza e mondo del lavoro.

Dal punto di vista dei contenuti il percorso ruoterà intorno alle seguenti aree tematiche: architettura dei computer (AC), sistemi operativi (SO), algoritmi e linguaggi di programmazione (AL), elaborazione digitale dei documenti (DE), reti di computer (RC), struttura di Internet e servizi (IS), computazione, calcolo numerico e simulazione (CS), basi di dati (BD).

Competenze

Al termine del percorso liceale lo studente deve:

- padroneggiare i più comuni strumenti software per il calcolo, la ricerca e la comunicazione in rete, la comunicazione multimediale, l'acquisizione e l'organizzazione dei dati;
- applicare tali strumenti in una vasta gamma di situazioni, ma soprattutto nell'indagine scientifica, scegliendo di volta in volta lo strumento più adatto in relazione all'analisi dei dati e alla modellizzazione del problema;
- avere una sufficiente padronanza di uno o più linguaggi per sviluppare applicazioni semplici, ma significative, di calcolo in ambito scientifico;
- scegliere i componenti hardware e software più adatti alle diverse situazioni e le loro configurazioni, valutarne le prestazioni e mantenerli in efficienza;
- possedere i principi scientifici del calcolo aritmetico ed algebrico, di individuazione delle strategie appropriate per la soluzione dei problemi e di analisi dei dati e interpretazione degli stessi con metodi deduttivi, che stanno alla base delle strutture informatiche e delle loro applicazioni;
- valutare l'opportunità, i limiti, le problematiche socio-culturali e i rischi dell'uso degli strumenti informatici.

INFORMATICA. SECONDO ANNO

Obiettivi Specifici

Nel secondo anno verranno usati gli strumenti di lavoro più comuni del computer insieme ai concetti di base ad essi connessi.

Verranno introdotti gli elementi costitutivi di uno strumento di gestione di fogli di calcolo. (DE)

Verranno introdotti i principi alla base dei linguaggi di programmazione, illustrate le principali tipologie di linguaggi e il concetto di algoritmo, quale metodo ottimale di risoluzione di un problema. Verrà sviluppata la capacità di implementare un algoritmo con gli Schemi di Composizione Fondamentale per la costruzione di Diagrammi a Blocchi.

L'introduzione dei concetti di base di un linguaggio di programmazione imperativo, e la sintassi delle istruzioni di sequenza, selezione e iterazione, permetterà di descrivere semplici algoritmi strutturati con implementazioni del linguaggio di programmazione imperativo scelto. (AL)

Obiettivi Minimi

- Comprensione del testo.
- Riconoscimento di connessioni logiche.
- Uso di linguaggio appropriato.
- Uso degli strumenti per la produzione di semplici fogli di calcolo con campi calcolati e grafici cartesiani.
- Comprensione dei procedimenti che descrivono e trasformano l'informazione.
- Analisi di semplici problemi ed descrizione della soluzione mediante un corretto algoritmo risolutivo strutturato con SCF di Sequenza, Selezione, e Iterazione.
- Comprensione della relazione tra algoritmo e programma.
- Acquisizione del concetto di linguaggio di programmazione.
- Individuazione delle diverse fasi di realizzazione di un programma.
- Codifica di semplici programmi con variabili semplici, istruzioni di selezione e cicli.
- Codifica di semplici programmi con semplici Array.
- Conoscenza degli strumenti per lo sviluppo del software, ambienti IDE.

Indicazioni metodologiche

Dalla constatazione obiettiva che l'efficacia dell'intervento educativo didattico dipende in larga misura dalla motivazione e dal grado di coinvolgimento dello studente, saranno adottate le strategie più efficaci per stimolare la curiosità, la creatività e l'operosità degli studenti sollecitandoli ad assumere un atteggiamento critico e attivo nel proprio processo di apprendimento.

Un primo criterio metodologico nasce dal costatare che la funzione strumentale e l'identità culturale e tecnica dell'informatica sono fortemente connesse fra loro e vanno quindi trattate in modo integrato.

Un secondo criterio è la connessione permanente fra teoria e pratica. Il percorso quinquennale dovrebbe essere organizzato come una serie di passi in ciascuno dei quali si acquisiscono la conoscenza e la padronanza di uno strumento o di una classe di strumenti, la loro applicazione a problemi significativi, la scoperta dei concetti teorici ad essi sottostanti, la riflessione sui vantaggi e sui limiti e sulle conseguenze del loro uso.

Per dare senso all'uso di strumenti informatici occorre proporre problemi significativi e, nello stesso tempo, tali da permettere un collegamento permanente con le altre discipline.

È in questo modo che l'informatica, oltre a proporre i propri concetti e i propri metodi, diventa anche uno strumento del lavoro dello studente.

La creazione/utilizzazione di semplici simulazioni possibilmente connesse ad argomenti scientifici (studio quantitativo di una teoria, confronto di un modello con i dati, ecc.) consente di comprendere come funziona lo stretto rapporto fra l'informatica, le altre scienze e le loro applicazioni tecnologiche che è caratteristico della ricerca moderna.

Per quanto possibile, gli argomenti saranno introdotti in forma di situazioni problematiche e gli studenti saranno sollecitati a riconoscere relazioni e a formulare ipotesi di soluzione facendo ricorso a conoscenze già acquisite e anche all'intuito; infine, attraverso procedimenti di tipo deduttivo, saranno guidati alla generalizzazione del risultato conseguito e alla sintesi con altre nozioni teoriche già apprese.

Le metodologie didattiche si concretizzeranno in termini di:

- **Situazioni di apprendimento**

Lezione frontale, apprendistato cognitivo, lezione interattiva/dialogica, approccio tutoriale, apprendimento basato su problema, caso, esercitazione, esercitazione di auto-correzione, collaborazione/cooperazione, approcci didattici individualizzati e di recupero per una più efficace partecipazione operativa degli alunni.

- **Materiale di supporto allo sviluppo dei contenuti**

Testi in adozione e/o consigliati, libri della biblioteca, presentazioni multimediali, documenti reperibili in rete, appunti integrativi del docente condivisi online, laboratorio con software di base e applicativi in dotazione al liceo, funzionale alle attività programmate.

- **Strumenti di lavoro**

Quaderni, schede, fotocopie, lavagna tradizionale, lavagna interattiva multimediale LIM, computer, CD-ROM.

Strumenti di verifica

Le verifiche sistematiche e periodiche saranno articolate in riferimento agli obiettivi generali e agli obiettivi specifici per ogni singolo argomento o unità didattica.

Per l'area cognitiva le prove saranno predisposte secondo i seguenti livelli di specificazione:

1. Conoscenza dei termini
2. Conoscenza degli argomenti
3. Conoscenza di regole
4. Capacità di effettuare trasformazioni e adattamenti
5. Capacità di stabilire relazioni

Si avrà cura inoltre di somministrare prove a vari livelli di complessità per consentire ad ognuno di dare risposte adeguate alle proprie capacità, tenendo conto non solo delle esigenze di chi ha particolari difficoltà, ma anche di quelle di chi dimostra maggiori abilità e più vivo interesse.

Le verifiche scritte e orali saranno frequenti e omogeneamente distribuite nell'arco dell'anno in accordo con quanto deciso in sede di collegio dei docenti.

Le prove scritte saranno articolate nelle forme più varie, dalle tipologie più tradizionali (esercizi di applicazione, analisi di problemi e conseguente sviluppo dell'algoritmo e poi del programma nel linguaggio di programmazione studiato) ai test e alle prove semi-strutturate.

Le interrogazioni orali mireranno soprattutto a valutare le capacità di ragionamento, di rielaborazione personale e di comunicazione attraverso un linguaggio proprio, chiaro e corretto.

Le verifiche pratiche verranno effettuate nell'ambito del laboratorio di Informatica e saranno articolate nello sviluppo di esercizi/algoritmi/semplici programmi.

Criteri di valutazione

La valutazione è un processo che tiene conto di tutti gli obiettivi presenti nella programmazione di dipartimento. Si ritiene tuttavia di sottolineare che, in relazione agli obiettivi per i singoli nuclei, si osserverà la capacità dell'allievo di:

- conoscere i contenuti dei diversi nuclei
- applicare in modo corretto i vari formalismi e metodi, e le varie tecniche
- analizzare un quesito e rispondere in forma sintetica
- prospettare soluzioni, verificarle e formalizzarle nonché l'aderenza ad alcuni obiettivi trasversali, fra i quali:
 - leggere e interpretare un testo di carattere scientifico
 - comunicare e formalizzare procedure
 - rielaborare in modo personale e originale i contenuti
 - partecipare in modo costruttivo e critico alle lezioni
 - acquisire un metodo di lavoro adeguato rispetto agli obiettivi prestabiliti
 - progredire nell'apprendimento rispetto ai livelli di partenza

L'enunciazione delle griglie, nel corpo dei testi delle prove, è comunque un ulteriore elemento a supporto di una valutazione efficace e leggibile.

Per la valutazione delle prove scritte

Per la correzione della prova scritta si terrà conto di quattro indicatori ai quali verranno attribuiti dei pesi differenti in base alla tipologia dell'esercizio e del numero dei quesiti proposti nella verifica come nell'esempio riportato.

Il punteggio verrà poi trasferito in un voto in decimi in base ad una articolazione che assegna la sufficienza nel caso di raggiungimento degli obiettivi minimi e in ogni caso viene comunicato e formalizzato alla riconsegna della prova.

Segue griglia per la valutazione prova scritta.

Indicatori	%	Valore massimo attribuibile 100					
		Q1	Q2	Q3	Q4	Q5	Q6
COMPRENSIONE e CONOSCENZA <i>Comprende la richiesta.</i> <i>Conosce i contenuti.</i>	30	(0-3)	(0-4)	(0-5)	(0-4)	(0-5)	(0-5)
ABILITA' LOGICHE e RISOLUTIVE <i>È in grado di separare gli elementi dell'esercizio evidenziandone i rapporti.</i> <i>Usa un linguaggio appropriato.</i> <i>Sceglie strategie risolutive adeguate, procedure ottimali, appropriate, originali e/o logicamente più valide.</i>	30	(0-3)	(0-4)	(0-5)	(0-4)	(0-5)	(0-5)
CORRETTEZZA dello SVOLGIMENTO <i>Organizzazione di algoritmi risolutivi corretti. Applica tecniche e procedure, anche grafiche, corrette.</i>	20	(0-2)	(0-4)	(0-5)	(0-4)	(0-5)	(0-5)
ARGOMENTAZIONE <i>Giustifica e commenta le scelte effettuate con chiarezza e in modo ordinato e strutturato.</i>	20	(0-2)	(0-3)	(0-5)	(0-3)	(0-5)	(0-5)
Punteggio totale quesito		(0-10)	(0-15)	(0-20)	(0-15)	(0-20)	(0-20)

Punteggio conseguito/100 _____ **Voto/10** _____

Tabella di conversione in base 10.

Punti	< 20	20-29	30-39	40-49	50-59	60-69	70-79	80-89	90-100
Voto	2	3	4	5	6	7	8	9	10

Per la valutazione delle prove orali

Per la valutazione delle interrogazioni ci si atterrà allo schema seguente, che ha la funzione di correlare i voti assegnati con un insieme di descrittori.

Livello	Descrittori	Voto
Gravemente insufficiente	Conoscenze estremamente frammentarie; gravi errori concettuali; palese incapacità di avviare procedure di astrazione, analisi, progettazione e codifica; linguaggio ed esposizione inadeguati.	1-3 /10
Decisamente insufficiente	Conoscenze molto frammentarie; errori concettuali; scarsa capacità di gestire procedure di analisi, di progetto e di codifica; incapacità di stabilire collegamenti, anche elementari; linguaggio inadeguato.	3-4 /10
Insufficiente	Conoscenze frammentarie, non strutturate, confuse; modesta capacità di gestire procedure di analisi, di progetto e di codifica; difficoltà nello stabilire collegamenti fra contenuti; linguaggio non del tutto adeguato.	4-5 /10
Non del tutto sufficiente	Conoscenze modeste, viziate da lacune; poca fluidità nello sviluppo e controllo delle procedure di analisi, di progetto e di codifica; applicazione di tecniche e metodi in forma mnemonica, insicurezza nei collegamenti; linguaggio accettabile, non sempre adeguato.	5-6 /10
Sufficiente	Conoscenze adeguate, pur con qualche imprecisione; padronanza di procedure di analisi, progetto e/o codifica, anche con qualche lentezza e capacità di gestire e organizzare procedure se opportunamente guidato; linguaggio accettabile.	6 /10
Discreto	Conoscenze omogenee e ben consolidate; padronanza di procedure di analisi, progetto e/o codifica, capacità di previsione e controllo; capacità di applicazione delle tecniche e dei metodi; autonomia nell'ambito di semplici ragionamenti; linguaggio adeguato e preciso.	6-7 /10
Buono	Conoscenze solide, assimilate con chiarezza; fluidità nel calcolo; autonomia di collegamenti e di ragionamento e capacità di analisi, progettazione logica e codifica; riconoscimento di schemi, adeguamento di procedure esistenti; individuazione di semplici strategie di risoluzione e loro formalizzazione; buona proprietà di linguaggio.	7-8 /10
Ottimo	Conoscenze ampie e approfondite; capacità di analisi e rielaborazione personale; fluidità ed eleganza nell'applicazione di tecniche e metodi, possesso di dispositivi di controllo e di adeguamento delle procedure; capacità di costruire proprie strategie di risoluzione; linguaggio sintetico ed essenziale.	8-9 /10
Eccellente	Conoscenze ampie, approfondite e rielaborate, arricchite da ricerca e riflessione personale; padronanza e eleganza nell'applicazione di formalismi; disinvoltura nel costruire proprie strategie di risoluzione, capacità di sviluppare e comunicare risultati di una analisi in forma originale e convincente.	9-10 /10

Informatica. Sostegno-Potenziamento-Recupero

Durante le ore di lezione saranno seguiti in particolare gli studenti in difficoltà e saranno corretti, anche individualmente, gli esercizi risolti a casa. Interventi mirati sia al recupero di abilità specifiche e sia all'acquisizione di un più adeguato metodo di studio.

Si privilegerà il recupero in itinere che verrà svolto dopo il primo quadrimestre.

Per vivacizzare l'interesse e la partecipazione costruttiva degli alunni più dotati, essi saranno costantemente impegnati in esercitazioni a più elevati livelli di complessità e in attività integrative di approfondimento.

Sarà incoraggiata la partecipazione a: concorsi e gare disciplinari [Olimpiadi dell'Informatica, MediaShow]

Informatica. Definizione dei Contenuti

Vengono riportate le articolazioni in moduli e l'insieme dei contenuti per ogni modulo.

Per ogni nucleo vengono indicate alcune prestazioni attese, e un insieme di contenuti ragionevolmente correlato a tali prestazioni.

I moduli vengono riportati cercando di rispettare un possibile ordine storico-propedeutico.

Modulo 6. Introduzione alla programmazione

(AL 4) SCF di iterazione

- | | |
|--|---|
| <ul style="list-style-type: none"> • Essere in grado di individuare le strutture più idonee alla modellizzazione di uno specifico problema iterativo. | <ul style="list-style-type: none"> • Il Ciclo pre-condizionale il ciclo post-condizionale • I cicli strutturati. • Il ciclo a conteggio. • Tecnica del contatore incrementato • Tecnica delle somme successive |
|--|---|

(AL 5) Linguaggi e strumenti per la programmazione

- | | |
|--|--|
| <ul style="list-style-type: none"> • Conoscere le tipologie di linguaggio di programmazione. • Implementare un programma strutturato in un linguaggio di programmazione per la risoluzione di una classe di problemi. • Conoscere le fasi e quali strumenti software usare per la codifica. • Valutare il software in termini di efficienza, efficacia e qualità. • Produrre un'efficace documentazione contestualmente alle varie fasi di sviluppo del software. | <ul style="list-style-type: none"> • I linguaggi di programmazione e loro classificazione |
| | <ul style="list-style-type: none"> • I programmi traduttori: compiler, interpreti, assembleri |
| | <ul style="list-style-type: none"> • Le fasi della codifica: sviluppo ed esecuzione di un programma |
| | <ul style="list-style-type: none"> • La messa a punto di un programma debugging |

(AL 6) I pacchetti IDE

- | | |
|--|---|
| <ul style="list-style-type: none"> • Utilizzare un ambiente IDE per la scrittura, la compilazione ed l'esecuzione di un programma. Riconoscere i principali errori segnalati dal compilatore. Utilizzare i debugger per individuare gli errori. | <ul style="list-style-type: none"> • I pacchetti IDE |
|--|---|

Modulo 7. Programmare con un linguaggio imperativo (ad es. linguaggio C/C++)	
(AL 7) Concetti di base di un linguaggio imperativo: variabili, istruzioni di selezione,	
<ul style="list-style-type: none"> Utilizzare le variabili nei programmi Scrivere programmi con istruzioni in sequenza e in blocchi. Scrivere codice con selezioni e iterazioni Utilizzare la simulazione dell'esecuzione di un programma e simulare la situazione delle variabili in memoria. Saper individuare errori nel codice 	<ul style="list-style-type: none"> Variabili semplici, tipi di dati ed espressioni Istruzioni if, switch Variabili booleane, operatori di confronto ed operatori logici Istruzioni for, while e do-while
(AL 8) Le variabili strutturate: gli Array	
<ul style="list-style-type: none"> Conoscere e manipolare le strutture dei dati array. Conoscere l'implementazione di un array. 	<ul style="list-style-type: none"> I vettori e la loro manipolazione Le matrici e la loro manipolazione Implementazione delle strutture dati
(AL 9) Gli algoritmi fondamentali (ricerca, ordinamento)	
<ul style="list-style-type: none"> Saper scegliere l'algoritmo adeguato alla situazione. 	Gli algoritmi fondamentali (ricerca, ordinamento)
(DE 5) Modulo 8. L'elaborazione di documenti elettronici: fogli di calcolo	
<ul style="list-style-type: none"> Raccogliere, organizzare e rappresentare un insieme di dati. Riconoscere una relazione tra variabili e la formalizzazione attraverso una funzione matematica. Esprimere il risultato di una misura attraverso un intervallo di valori e stabilire l'errore assoluto/relativo. Definire fogli di calcolo con campi calcolati e grafici cartesiani. 	<ul style="list-style-type: none"> L'elaborazione di documenti elettronici: fogli di calcolo

SCANSIONE DEI CONTENUTI (MODULI) DEL PIANO DI STUDIO

Informatica. Classe Seconda

I Quadrimestre	Settembre - Ottobre - Novembre	<ul style="list-style-type: none">• Ripetizione o eventuale completamento del programma dell'anno precedente• Introduzione alla programmazione<ul style="list-style-type: none">- SCF di Iterazione- Linguaggi e strumenti di programmazione- I Pacchetti IDE• Programmare con un Linguaggio Imperativo<ul style="list-style-type: none">- Concetti di base: variabili semplici, tipi di dati ed espressioni,
	Dicembre - Gennaio	<ul style="list-style-type: none">• Programmare con un Linguaggio Imperativo<ul style="list-style-type: none">- Istruzioni if, while e do-while
II Quadrimestre	Febbraio - Marzo - Aprile	<ul style="list-style-type: none">• Programmare con un Linguaggio Imperativo<ul style="list-style-type: none">- Istruzione for- Gli Array- Gli Algoritmi fondamentali
	Maggio - Giugno	<ul style="list-style-type: none">• L'elaborazione con fogli di calcolo