

PROGRAMMAZIONE

Materia	INFORMATICA		
Classe	4A LATTANZIO	Anno scolastico	2018-19
Libro di testo			
Autore	LORENZI AGOSTINO / MORIGGIA VITTORIO / RIZZI ANDREA		
Titolo	INFORMATICA PER ISTITUTI TECNICI TECNOLOGICI vol, B	ISBN	9788826818399
Casa Editrice	ATLAS	Prezzo	18,80

DOCENTI	Insegnante teorico	PAOLA PASCUCCI	
	Insegnante tecnico pratico	RAFFAELE GRIPPA	

RISULTATI DI APPRENDIMENTO ATTESI:

In riferimento alle linee guida si elencano i risultati di apprendimento individuati per la materia:

Competenze:

- Riconoscere le situazioni in cui utilizzare il paradigma a oggetti
- sviluppare applicazioni informatiche per reti locali o servizi a distanza;
- saper leggere e saper documentare le applicazioni con linguaggio UML
- Utilizzare a livello base la lingua inglese per la consultazione di manuali tecnici del settore
- Utilizzare le risorse in rete per la ricerca autonoma di soluzioni ad eventuali problemi applicativi
- Collaborare al raggiungimento degli obiettivi all'interno del gruppo partecipando all'analisi e alla realizzazione delle soluzioni dei problemi proposti

Conoscenze:

- Sviluppare applicazioni informatiche con programmazione guidata dagli eventi e interfaccia grafica
- Paradigma di programmazione a oggetti
- Linguaggio di programmazione lato client per la gestione locale di eventi in pagine web.
- Lessico e terminologia tecnica di settore anche in lingua inglese

Abilità:

- Progettare e implementare applicazioni secondo il paradigma ad oggetti.
- Progettare e realizzare interfacce grafiche utente
- Progettare e realizzare pagine web statiche con interazione locale.
- Utilizzare il lessico e la terminologia tecnica di settore anche in lingua inglese

ARTICOLAZIONE ORARIA

Sono previste 3 ore di teoria e 3 ore di esercitazioni in laboratorio

ANALISI DELLA SITUAZIONE DI PARTENZA DELLA CLASSE

Profilo generale della classe

La classe è composta da 14 alunni di cui 1 ripetente di questo stesso Istituto e uno proveniente da altro istituto. Gli altri studenti appartengono allo stesso gruppo classe dello scorso anno, pertanto gli alunni si conoscono e non ci sono rapporti conflittuali tra di loro.

Alunni con disturbi specifici dell'apprendimento:

Non risultano alunni con difficoltà di apprendimento .

Livelli di partenza rilevati

Conoscendo la classe, si deve purtroppo affermare che per alcuni le lacune accumulate lo scorso anno e non adeguatamente colmate, si ripercuoteranno indubbiamente sul profitto dell'anno in corso, essendo gli argomenti della disciplina in oggetto una prosecuzione e un approfondimento di quanto appreso lo scorso anno.

Sono sempre presenti alcuni alunni che già mostravano interesse e impegno lo scorso anno e che mostrano maturando un atteggiamento un po' più maturo e responsabile.

L'alunno proveniente da altro istituto non ha le conoscenze di base adeguate per affrontare lo studio degli argomenti della disciplina del quarto anno.

UNITÀ DISCIPLINARI DEL PERCORSO FORMATIVO

Le unità didattiche prevedono lezioni teoriche e lezioni pratiche di laboratorio

Il percorso formativo segue la programmazione di dipartimento a cui si fa riferimento per la definizione degli obiettivi minimi di apprendimento e nuclei essenziali del programma.

Modulo 0	Ripasso Argomenti propedeutici Anno precedente
Periodo: settembre	<ul style="list-style-type: none"> • Funzioni , parametri e valori di ritorno • Strutture fondamentali di un algoritmo • Operatori aritmetici , di confronto e booleani • Identificatori; variabili; costanti • Tipi di dato; operatori; commenti e documentazione

Modulo 1	COMPETENZE	
Programmazione orientata agli oggetti	<ul style="list-style-type: none"> • Descrivere i concetti di base della programmazione ad oggetti • Individuare gli aspetti della metodologia orientata agli oggetti 	
Periodo	Conoscenze	Abilità
Ottobre Novembre	<ul style="list-style-type: none"> • Una nuova metodologia • Orientamento agli oggetti: oggetti e classi • Dichiarazione e utilizzo di una classe • Attributi e metodi • Creazione e utilizzo di oggetti • Mascheramento dell'informazione • Livelli di visibilità • Ereditarietà • Dichiarazione e utilizzo di sottoclassi • Gerarchia delle classi • Polimorfismo 	<ul style="list-style-type: none"> • Saper individuare i passi della metodologia orientata agli oggetti • Definire classi con attributi e metodi • Disegnare i diagrammi di classe in UML • Conoscere i diagrammi UML per la rappresentazione di Classi, Oggetti, attributi e Metodi

Modulo 2 Ambiente di programmazione java		COMPETENZE <ul style="list-style-type: none"> • Individuare gli aspetti caratterizzanti di un nuovo linguaggio di programmazione • Progettare la struttura generale di un programma • Utilizzare correttamente la sintassi e i costrutti del linguaggio 	
Periodo	Conoscenze	Abilità	
Ottobre Novembre Dicembre	<ul style="list-style-type: none"> • Caratteristiche generali • L'ambiente di programmazione • La struttura di un'applicazione • Identificatori; variabili; costanti • Tipi di dato; operatori; commenti e documentazione • Le strutture di controllo dell'algoritmo • Compilazione e interpretazione di un linguaggio Java • Operazioni su standard input e standard output • Array ad una e due dimensioni • Array di oggetti • Creazione di oggetti • Applicazione dell'ereditarietà e del polimorfismo agli oggetti • Manipolazione di stringhe • Eccezioni 	<ul style="list-style-type: none"> • Creare e utilizzare oggetti • Saper scrivere , compilare ed eseguire un programma Java • Saper rappresentare le operazioni di I/O standard • Saper dichiarare le variabili e le costanti • Saper inserire frasi di commento nel programma • Saper utilizzare le strutture di controllo • Saper gestire semplici eccezioni • Dichiarazione, allocazione e utilizzo di array mono e bidimensionali • Saper utilizzare i livelli di visibilità di attributi e metodi • Saper creare sottoclassi applicando l'ereditarietà • Saper applicare l'overriding e l'overloading agli oggetti • Saper manipolare stringhe 	

Modulo 3 Strutture dinamiche e gestione dei file		COMPETENZE <ul style="list-style-type: none"> • Analisi di strutture di tipo pila e/o coda • Memorizzare dati su supporti di memoria di massa 	
Periodo	Conoscenze	Abilità	
Febbraio Marzo	<ul style="list-style-type: none"> • Introduzione; array dinamici • Gestione automatica della memoria • Modalità di gestione delle strutture: • pila, coda, lista, albero binario • I flussi di input/output • File strutturati • File di testo 	<ul style="list-style-type: none"> • Saper individuare le caratteristiche delle strutture dinamiche • Conoscere le tecniche di gestione delle strutture dati dinamiche • Distinguere i diversi tipi di file • Gestire le diverse modalità di accesso ai file utilizzando il linguaggio Java 	

Modulo 4 Interfaccia grafica in java		COMPETENZE <ul style="list-style-type: none"> • Comprendere l'importanza della progettazione dell'interfaccia per l'utente • Costruire interfacce per l'utente usando gli elementi grafici fondamentali 	
Periodo	Conoscenze	Abilità	

Febbraio Marzo Aprile	<ul style="list-style-type: none">• Caratteristiche generali dell'interfaccia per l'utente; interfacce a carattere e interfacce grafiche• Gli oggetti dell'interfaccia grafica• Le librerie AWT e SWING• I contenitori e le componenti• La disposizione degli elementi grafici• La gestione degli eventi	<ul style="list-style-type: none">• Saper progettare semplici interfacce usando gli elementi grafici fondamentali• Creare i contenitori finestra e pannello• Creare le componenti da inserire nei contenitori• Usare i gestori del layout per disporre gli elementi grafici• Saper gestire gli eventi associati ai principali elementi grafici
--	---	--

Modulo 5 Linguaggi web lato client		COMPETENZE
		<ul style="list-style-type: none"> Progettare , realizzare e gestire pagine Web statiche e dinamiche lato client
Periodo	Conoscenze	Abilità
Aprile Maggio	Applet <ul style="list-style-type: none"> Applicazioni e applet Il linguaggio HTML: elementi fondamentali La classe JApplet I parametri Applet e browser Interazioni con il browser Disegni Immagini nelle applet Il linguaggio Javascript <ul style="list-style-type: none"> Caratteristiche principali del linguaggio JavaScript Gestione degli eventi in JavaScript 	<ul style="list-style-type: none"> Utilizzare le componenti grafiche per costruire interfacce nelle pagine Web Costruire applet con disegni e immagini Scrivere pagine Web statiche Inserire uno script JavaScript nella pagina HTML Organizzare le istruzioni JavaScript in funzioni Scrivere il codice per la gestione degli eventi Gestire l'interazione con l'utente Immagini

OBIETTIVI MINIMI (comuni a tutte le classi parallele, individuati dal Dipartimento)

Il percorso formativo segue la programmazione di dipartimento a cui si fa riferimento per la definizione degli obiettivi minimi di apprendimento e nuclei essenziali del programma

Si considerano noti gli argomenti scritti nel modulo 0.

Si ritiene fondamentale il conseguimento degli obiettivi minimi per il passaggio alla classe successiva

CONOSCENZE	ABILITA'
Il linguaggio Java <ul style="list-style-type: none"> Orientamento agli oggetti: oggetti e classi Dichiarazione e utilizzo di una classe Attributi e metodi Creazione e utilizzo di oggetti Livelli di visibilità Mascheramento dell'informazione, Ereditarietà Dichiarazione e utilizzo di sottoclassi Gerarchia delle classi Polimorfismo La struttura di un'applicazione Java Compilazione e interpretazione di un linguaggio Java Operazioni su standard input e standard output Array ad una dimensione Array di oggetti Eccezioni 	<ul style="list-style-type: none"> Saper individuare i passi della metodologia orientata agli oggetti Definire classi con attributi e metodi Disegnare i diagrammi di classe in UML Conoscere i diagrammi UML per la rappresentazione di Classi, Oggetti, attributi e Metodi Creare e utilizzare oggetti Saper scrivere , compilare ed eseguire un programma Java Saper dichiarare le variabili e le costanti Saper inserire frasi di commento nel programma Saper utilizzare le strutture di controllo Saper gestire semplici eccezioni Dichiarazione, allocazione e utilizzo di array monodimensionali Dichiarazione, allocazione e utilizzo di array di oggetti Saper utilizzare i livelli di visibilità di attributi e

<ul style="list-style-type: none">• Gestione automatica della memoria• array dinamici• pila, coda, lista, albero binario• Caratteristiche generali dell'interfaccia grafica• Gli oggetti dell'interfaccia grafica• I contenitori e le componenti• La disposizione degli elementi grafici• La gestione degli eventi	<p>metodi</p> <ul style="list-style-type: none">• Saper creare sottoclassi applicando l'ereditarietà• Saper applicare l'overriding e l'overloading agli oggetti• Saper manipolare stringhe• Saper individuare le caratteristiche delle strutture dinamiche• Conoscere le tecniche di gestione delle strutture dati dinamiche• Saper progettare semplici interfacce usando gli elementi grafici fondamentali• Creare i contenitori finestra e pannello• Creare le componenti da inserire nei contenitori• Usare i gestori del layout per disporre gli elementi grafici• Saper gestire gli eventi associati ai principali elementi grafici
<p>Il linguaggio Javascript e HTML</p> <ul style="list-style-type: none">• Caratteristiche principali del linguaggio JavaScript• Gestione degli eventi in JavaScript	<ul style="list-style-type: none">• Utilizzare le componenti grafiche per costruire interfacce nelle pagine Web• Scrivere pagine Web statiche• Inserire uno script JavaScript nella pagina HTML• Organizzare le istruzioni JavaScript in funzioni• Scrivere il codice per la gestione degli eventi• Gestire l'interazione con l'utente

FORMATI DIDATTICI PREVISTI

- Lezione frontale e interattiva con utilizzo della LIM.
- Tecnica di problem solving per mettere gli studenti di fronte a situazioni aperte che vengono affrontate per tentativi anche empirici.
- Gruppi di lavoro
- Attività pratica di laboratorio in cui trovano adeguata verifica le metodologie sopra elencate

Le lezioni si svolgeranno in aula e in laboratorio di informatica come previsto dall'orario scolastico.

STRUMENTI E MATERIALI DIDATTICI

- Libro di testo
- schede di lavoro proposte dai docenti
- eventuali dispense di approfondimento e/o completamento degli argomenti proposti
- Internet per la ricerca autonoma di soluzioni a quesiti proposti

TIPOLOGIA DELLE PROVE DI VERIFICA PREVISTE

Le prove di verifica che si intende utilizzare sono:

- Prove pratiche (laboratorio): lavori individuali e/o di gruppo, relazioni individuali.
- Prove scritte di diverso tipo:
 1. quesiti a risposta singola. Le risposte devono essere autonomamente formulate dallo studente;
 2. quesiti con risposta a scelta multipla. L'alunno deve scegliere tra più opzioni la risposta giusta
 3. realizzazione di applicazioni Object Oriented , rappresentazione con diagramma UML codifica in linguaggio di programmazione Java
- Prove orali: interrogazioni, interventi significativi e partecipazione al dialogo educativo.

Numero **minimo** di prove per quadrimestre: **4** (1 pratica, 2 scritte, 1 orale).

CRITERI DI VALUTAZIONE, INDICATORI E GRIGLIE

Per i criteri di valutazione ci si atterrà a quelli illustrati nel P.T.O.F.

Gli indicatori di riferimento sono:

per la produzione scritta

- a) correttezza e proprietà di linguaggio
- b) pertinenza alla traccia
- c) conoscenza dei contenuti

- d) coerenza logica
- e) rielaborazione personale

per il colloquio:

- a) correttezza nell'uso dei linguaggi specifici
- b) conoscenza degli argomenti e capacità di individuarne gli elementi fondanti

- c) capacità di effettuare collegamenti
- d) capacità di esprimere giudizi motivati

per l'attività di laboratorio:

- a) cooperazione
- b) autonomia

- c) responsabilità
- d) rispetto delle consegne

Per le prove scritte e le prove pratiche di laboratorio si seguiranno le seguenti regole:

- ogni quesito sarà accompagnato da uno specifico punteggio stabilito dal docente per i casi di quesito corretto/errato/assente
- I punti totali ottenuti dallo studente saranno tradotti in punti decimi secondo una scala pubblicata dal docente contestualmente alla prova

Per lo scritto che richiede la produzione di un programma, verranno presi in considerazione i seguenti elementi (con relativa incidenza sul voto finale):

- | | |
|--|-----|
| • corretta e completa individuazione delle proprietà della classe | 20% |
| • corretta individuazione e realizzazione dei metodi della classe | 40% |
| • gradevolezza dell'interfaccia (a caratteri o grafica) progettata | 20% |
| • documentazione, utilizzo di commenti | 15% |
| • originalità della soluzione e/o ottimizzazioni introdotte | 5% |

Laddove non presente uno o più degli elementi valutativi precedenti, la relativa percentuale verrà ripartita sulle altre componenti.

Per le interrogazioni orali verranno presi in considerazione i seguenti elementi (con relativa incidenza sul voto finale):

- | | |
|---|-----|
| • padronanza del linguaggio tecnico | 30% |
| • conoscenza degli argomenti (livello di approfondimento e capacità di cogliere l'essenza del tema esposto) | 50% |
| • capacità di orientamento autonomo, cogliere nessi e operare raccordi | 20% |

I voti saranno espressi in decimi secondo l'intervallo approvato dal Collegio dei Docenti

ATTIVITÀ DI RECUPERO, SOSTEGNO, APPROFONDIMENTO

In itinere saranno svolte attività di recupero a seconda delle necessità. Durante le settimane di recupero deliberate del Collegio docenti e definite dal Consiglio di Classe, è prevista la diversificazione dell'attività didattica, nelle ore di compresenza, con interventi di sostegno per gli studenti in difficoltà e di approfondimento per gli altri. Considerando che le principali difficoltà riscontrate fanno riferimento alla comprensione del testo e all'individuazione della strategia risolutiva dei problemi, si proporrà un'attività di *“laboratorio di problem solving”*.

ATTIVITÀ DI ALTERNANZA SCUOLA LAVORO

La programmazione disciplinare potrebbe subire delle modifiche in base alle attività che la classe dovrà svolgere per l'Alternanza Scuola Lavoro. Si aderirà alle iniziative proposte dalla commissione ASL e approvate dal Consiglio di Classe.

RAPPORTI CON LE FAMIGLIE

Gli incontri con le famiglie avverranno con il ricevimento in orario antimeridiano e i 2 incontri generali pomeridiani stabiliti in sede di programmazione delle attività scolastiche.