



Distretto Scolastico n. 2

ISTITUTO di ISTRUZIONE SUPERIORE “Ettore Majorana”

Via Aldo Moro 1, 67051 Avezzano (AQ) - Tel. 086322570

ISTITUTO TECNICO settore TECNOLOGICO: Elettronica ed Elettrotecnica; Informatica e Telecomunicazioni; Meccanica, Meccatronica ed Energia; Chimica, materiali e biotecnologie.

ISTITUTO PROF.LE SETTORE INDUSTRIA E ARTIGIANATO: Manutenzione e Assistenza Tecnica
Cod.Mecc.:AQIS01400c - Cod. Fisc. 81005180666 - Web: www.iisavezzanomajorana.it - e-mail: aqis01400c@istruzione.it

PROGRAMMA SVOLTO

Anno scolastico 2022/2023

DISCIPLINA: Scienze e Tecnologie Applicate

CLASSE 2 Sez. F

INDIRIZZO Informatica e Telecomunicazioni

Programma Svolto

Di seguito si riportano le conoscenze che gli alunni hanno acquisito, i contenuti utilizzati e le relative abilità sviluppate, con scansione temporale.

MODULO (M)	PERIODO
M0 - Introduzione	Trimestre
M1 - Problemi, algoritmi e programmazione + Programmazione C++ e Python	Trimestre/Pentamestre
M2 - Elementi di matematica per l'Informatica	Trimestre

Trimestre

M	CONOSCENZE	CONTENUTI	ABILITA'
M0		- Breve storia dell'informatica e delle telecomunicazioni	
M1	- Significato di comunicazione con il computer - Varie Categorie di linguaggi - Significato di algoritmo, caratteristiche e rappresentazione - Significato di dato e principali tipi di dati - Ambiente C++ - Struttura di un programma C++	- Programmare e comunicare - La realizzazione di un programma - I principali linguaggi di programmazione - Descrizione, caratteristiche e rappresentazione di un algoritmo - Dati di input, output e di lavoro - Variabili e costanti - Selezione e cicli - Array - Ambiente C++	- Analizzare un problema e individuarne i vari tipi di dati - Risolvere un problema con un approccio algoritmo - Rappresentare un algoritmo con un diagramma di flusso - Costruire algoritmi che utilizzino sequenze, selezioni e iterazioni - Codificare un algoritmo in C++
M2	- Sistema di numerazione posizionale - Sistemi binario, ottale e esadecimale - Relazione tra i sistemi di numerazione	- I sistemi di numerazione posizionali: binario, ottale ed esadecimale - La relazione tra i vari sistemi di numerazione - Programma di conversione tra binario e decimale e viceversa	- Esprimere un numero in notazione posizionale - Operare nei vari sistemi di numerazione - Convertire numeri da un sistema a un altro

Pentamestre

M	CONOSCENZE	CONTENUTI	ABILITA'
M2	- Sistema di numerazione posizionale - Sistemi binario, ottale e esadecimale - Relazione tra i sistemi di numerazione - Elementi di base dell'algebra booleana: definizione e tavole di verità - Principali connettivi logici - Proposizione ed espressione logica - Proprietà e teoremi dell'algebra booleana	- I sistemi di numerazione posizionali: binario, ottale ed esadecimale - La relazione tra i vari sistemi di numerazione - Programma di conversione tra binario e decimale e viceversa - Elementi e applicazioni di algebra booleana - Le proprietà dell'algebra di Boole e la trasformazione di espressioni logiche	- Esprimere un numero in notazione posizionale - Operare nei vari sistemi di numerazione - Convertire numeri da un sistema a un altro - Costruire la tavola di verità di un'espressione logica - Dimostrare teoremi con le tavole di verità - Implementare un algoritmo di conversione in un linguaggio ad alto livello
M6	- Principali circuiti - Significato di rete logica - Forme canoniche di un'espressione logica - Concetto di circuito combinatorio - Concetto di circuito sequenziale	- I circuiti logici elementari e la rete logica - Analisi e sintesi di una rete combinatoria - Reti e circuiti sequenziali - Il sommatore binario completo: full adder	- Analizzare una rete - Sintetizzare una rete - Portare un'espressione logica in forma canonica - Descrivere semplici elementi in memoria - Descrivere i sommatore parziale e totale

M	CONOSCENZE	CONTENUTI	ABILITA'
M1	<ul style="list-style-type: none"> - Ambiente Python - Struttura di un programma Python 	<ul style="list-style-type: none"> - Programmare e comunicare - La realizzazione di un programma - I principali linguaggi di programmazione - Descrizione, caratteristiche e rappresentazione di un algoritmo - Dati di input, output e di lavoro - Variabili e costanti - Selezione e cicli - Array - Ambiente Python 	<ul style="list-style-type: none"> - Analizzare un problema e individuarne i vari tipi di dati - Risolvere un problema con un approccio algoritmo - Rappresentare un algoritmo con un diagramma di flusso - Costruire algoritmi che utilizzino sequenze, selezioni e iterazioni - Codificare un algoritmo in Python