

Programma Svolto AS 2025-2026 Saba Floris 3A Inf

Appunti, esercizi svolti, materiale didattico si trovano nella classroom. Molti esercizi svolti si trovano nel ramo di cartelle condivise raggiungibili a partire da Classroom.

Primo programma, in linguaggio C.

Significato di ciascuna riga del programma C (`#include <stdio.h>`, `int main() { }`, `return 0;`;

Output in C: Utilizzo di `printf` con stringhe e variabili e con specificatori di formato per i tipi interi, caratteri e stringhe.

Shell linux con i comandi `ls -l`, `pwd`, `clear`, `gcc`.

Compilazione ed Esecuzione di programmi.

Utilizzo base dell'editor vim: modalità inserimento e comando. Aprire, editare e salvare, uscire.

Linguaggio c: dichiarazione, inizializzazione e uso delle variabili di tipo intero. Uso della funzione `sizeof(tipo)` e dei qualificatori `short`, `long` e `unsigned` per le variabili intere. Esempi in linguaggio C.

Ambienti per programmare in C in windows: `dev-c++` (<https://www.bloodshed.net>), installazione windows subsystem for linux in windows.

Linguaggio C: uso di `printf` con variabili di tipo `int`, `float` e `double`.

Input in C: uso di `scanf` con `int`, `float` e `double`.

Assegnazione e operatori matematici: `+`, `-`, `*`, `/` e `%` (somma, sottrazione, moltiplicazione, divisione, resto solo su interi).

Procedimento/metodologia per la risoluzione di un problema informatico. Esempio di scrittura di un programma che risolve il problema dato (all'interno dell'esercizio `interi.c`)

Operatori matematici del c in particolare `/` e `%` (divisione intera e resto)

Operatore indirizzo `&`. Specificatore di formato per gli indirizzi o puntatori `%p`

Tipi `char` e `string` (vettori di caratteri) dichiarazione, acquisizione e stampa.

Tranelli ed accortezze nell'uso di `scanf()` per l'input in C.

Stringhe in C. Dichiarazione di variabili stringhe (array di caratteri). Assegnazione di stringhe in dichiarazione. File include per le funzioni sulle stringhe: funzioni `strcpy` e `strlen`. Stampa di stringhe con `printf`. Acquisizione di stringhe da input con `scanf`. Rappresentazione in memoria delle stringhe in C: carattere terminatore di stringhe in c (`'\0'` o `0`)

Basi del linguaggio C direttive `#include`, `#define`, definizione variabili, `printf` e `scanf` con vari tipi di dato.

Metodologia di risoluzione problemi: esempio con risoluzione del problema della trasformazione dei gradi in secondi.

Operatori prefissi e postfissi di pre incremento e post incremento, pre decremento e post decremento.

Operatori compatti o ridotti di assegnazione (`+=`, `-=`, `*=`, `/=` e `%=`).

Struttura di sequenza in programmazione e istruzioni di controllo del flusso.

Istruzioni di **selezione unaria (o a una via)** e **binaria (o a due vie)** in pseudocodice, diagramma di flusso e linguaggio C.

Introduzione delle variabili di tipo booleano (bool in C): direttiva `#include <stdbool.h>`.

Istruzione di **iterazione (ciclo) a controllo in testa** (pseudocodice, diagramma di flusso e linguaggio C).

Classificazione istruzioni di iterazione definite e indefinite

Ciclo a controllo in coda o postcondizionale.

Istruzione di iterazione **enumerativa** (ciclo **for**)

Algebra booleana: operatori booleani **not**, **or** e **and**. Variabili booleane ed esempio banale di utilizzo, esempi con AND e NOT in diagramma di flusso e linguaggio C.

Numeri casuali in C. Librerie da includere `#include <stdlib.h>` e `#include <time.h>` e funzioni `srand(time(NULL))` e `rand()`. Range di numeri casuali generati e loro riduzione all'intervallo voluto.

Esercizi sui cicli con **contatori** e **accumulatori**, cicli con uscita con dato **tappo** e con **uscita su richiesta dell'utente**.

Istruzione di **selezione multipla (Switch-case-break)** in C.

Utilizzo variabile **accumulatore di prodotti** (esempi b^e prima con $e \geq 0$ e poi qualsiasi)

Tabella di traccia per la verifica della correttezza di algoritmi o piccole parti di algoritmo

Cicli annidati.

Introduzione ai **sottoprogrammi** e alle **funzioni** in C.

Operatore **ternario**

Simulazione di **procedure** in C (funzioni void).

Esempio della procedura che stampa una riga di caratteri in tre versioni.

Funzioni: significato dei **parametri formali** di una funzione e loro relazione con i **parametri attuali**. Esempi delle funzioni max e stampa riga.

Differenza tra procedura e funzione

Parametri formali ed attuali e loro differenza, regole sui nomi delle funzioni e significato ed importanza dei **prototipi**.

Parametri **per valore** e **per riferimento**: scambio di variabili con sottoprogramma.

Passaggio per **riferimento** e suoi effetti sui parametri attuali.

Esempio di scambio prima sbagliato (perché usa il passaggio per valore) e poi corretto usando il passaggio per indirizzo o per riferimento.

Esempio con i puntatori. Si svolge, spiegandolo nel dettaglio, un esercizio in cui si modifica una variabile modificando un puntatore che punta ad essa

Le funzioni **ricorsive**: definizione ed esempio di utilizzo con la funzione fattoriale.

Esercizi fattoriale 1-2-3 Con codifica della funzione fattoriale ricorsiva, funzione fattoriale iterativa e confronto dei tempi di esecuzione delle due implementazioni.

Array monodimensionali o vettori.

Significato e rappresentazione logica di un vettore. Dichiarazione ed inizializzazione di un vettore. Accesso in scrittura (assegnazione ad un elemento di un vettore) e lettura (utilizzo di un elemento di un vettore) ad un elemento di un vettore.

Strategia risolutiva dell'esercizio assegnato per casa sulle temperature massime e minime

I **vettori paralleli** esempi di uso di vettori paralleli.

Esempio calcolo delle frequenze di uscita di un dado lanciato tante volte. Inserimento di nomi e valutazioni di vario tipo in vettori paralleli e calcolo della media voto per ciascun alunno.

Funzioni **fgets**, **strlen** delle stringhe e **getchar()** (per pulire l'input)

Partendo dai due file mostrati a lezione gli alunni devono creare le funzioni e procedure adatte al problema. I due file sono nella cartella condivisa e su Classroom.

Array di stringhe e uso delle funzioni fgets, strlen e getchar.

L'ordinamento dei vettori: il **bubble sort**. Si mostra il codice del bubble sort ampiamente commentato e con stampe esplicative di quello che fa l'algoritmo. Il programma è nelle lezioni condivise/Lezioni/vettori/ordinamenti

Algoritmo di ordinamento **insertion Sort**

Esercizi su funzioni utili per comprendere poi arrivare all'algoritmo insertion sort
Introduzione alle strutture in C.

Le **matrici** in C. Esempi di accesso ai singoli elementi (lettura e scrittura), stampa matrice, inserimento righe o colonne, stampa diagonale principale e secondaria di matrici quadrate, stampa matrice triangolare superiore o inferiore.

Record (strutture in C). Definizione di strutture con campi semplici, definizione di strutture con campi di tipo struttura, definizione di strutture con campi di tipo Array.

Accesso ai campi di tipo struttura con operatore punto.

Puntatore a struttura, accesso ai campi di una struttura tramite puntatore.

Strutture e tabelle.

Esercizi

Qui sono riportati quasi tutti gli esercizi svolti in classe ed in laboratorio durante l'anno scolastico. Di parte di essi vi è una soluzione nelle cartelle condivise o su classroom

- Esempio di acquisizione dati con scanf e possibili problemi (file

tranelliConlChar.c)

- Scrivere un programma c che acquisisca la base e l'altezza di un rettangolo e calcoli perimetro e area
- dato nome e anno di nascita di una persona stampare il nome e l'età approssimativa di una persona
- Scrivere un programma che definisca delle stringhe adatte a contenere i seguenti concetti: il nome e il cognome di una persona (due variabili). Quindi acquisisca il nome e il cognome da input con la scanf e poi visualizzi i valori di tali variabili e la loro lunghezza. Il programma i seguito chieda all'utente di inserire un numero piccolo (minore della lunghezza della stringa) e quindi visualizzi il carattere della stringa i posizione i-esima (in pratica il carattere che ha come indice il numero indicato dall'utente)
- Scrivere un programma che definisca le variabili adatte a questa situazione: Si vuole acquisire in input il nome, il cognome, la data di nascita, e la squadra di calcio tifata da una persona e quindi stampino tutti questi dati
- Chiedere all'utente di inserire una stringa e calcolarne la lunghezza.
- Scrivere un programma che definisca variabili di tipo short, int, long e stampi la dimensione di ciascuna (operatore sizeof())
- Scrivere un programma che definisca una variabile di tipo stringa le assegni un valore e poi ne stampi la dimensione in byte
- Data l'area di una circonferenza, determinare il perimetro.
- Leggi il valore del raggio di una circonferenza e calcolane l'area e il perimetro.
- Scrivi un programma che, data in input la misura di un angolo in gradi(G), primi(P) e secondi(), determini la sua ampiezza espressa in secondi.
- Scrivere un programma che preso in input una lettera dell'alfabeto stampi a monitor la lettera stessa e le due lettere successive (N.B. i caratteri in C sono degli interi di un byte e ad essi si possono applicare le operazioni matematiche di somma e sottrazione
- Sapendo che i caratteri in C sono rappresentati secondo la tabella ASCII prendere in input una lettera minuscola e stampare sia la lettera presa in input che la corrispondente lettera maiuscola. Se non ricordate la tabella ASCII da terminale digitare, nella shell linux, man ascii
- Scrivi un programma che chieda 3 numeri e ne stampi la somma e il valore medio.
- Data la misura del perimetro di un quadrato, determinane l'area.
- Date le basi e l'altezza di un trapezio rettangolo, determinane l'area (non usate la formula diretta ma ragionate sfruttando i precedenti esercizi)
- Avendo il saldo del conto corrente in euro, calcolalo in lire e in dollari (in base al valore odierno del cambio).
- Dati i valori dell'ipotenusa e dei cateti di un triangolo rettangolo, calcolane il perimetro e l'area.

- Visualizza il numero precedente e il successivo di un numero intero letto in INPUT.
- Visualizza i tre numeri successivi di un numero naturale inserito in INPUT.
- Prendere in input nome della squadra e l'anno e stamparli in output
- prendere input un numero tra 1 e 10 quindi prendere in input carattere tra a e p, quindi sommare al carattere il numero e poi mostrarlo in output
- scambio di due variabili in linguaggio C Completare l'esercizio sullo scambio delle variabili in C
- Dati 3 numeri in input calcolare il massimo e il minimo (solo if semplici)
- Leggi tre numeri e individua se ci sono tra di essi numeri uguali
- Dati 3 numeri in input calcolare il massimo e il minimo (con if else)
- Dato un numero stabilire se è positivo o negativo.
- Dati 2 numeri stamparli in ordine crescente e decrescente.
- Dati due numeri e un carattere c o d stampare in ordine crescente o decrescente in base al carattere
- presi in input due numeri e memorizzati in n1 e n2, stampare prima n1 e n2 e poi dopo aver messo in n1 il più piccolo e in n2 il più grande stamparli nuovamente (scambio di variabili)
- presi in input tre numeri e memorizzati in n1, n2 e n3, stampare prima n1, n2 e n3 e poi dopo averli messi in ordine crescente stamparli nuovamente (scambio di variabili)
- Prendere due numeri interi in input e salvarli in due variabili n1 e n2. Mettere il numero più piccolo in n1 e il più grande in n2 (eventuale scambio di variabili). Stampare tutti i numeri compresi tra n1 e n2.
- stampare tutti i numeri compresi tra n1 e n2 dal più grande al più piccolo. n1 e n2 sono presi in input
- Scrivi un programma che invita l'utente a inserire da tastiera una sequenza
- di numeri finché la loro somma non supera il valore di 200, e visualizza a
- schermo quanti numeri sono stati inseriti.
- Calcolare il minimo tra N numeri presi in input
- Scrivi un programma che legge un numero NUM accertandosi che sia maggiore di 0 e visualizzi tutti i numeri pari inferiori a quel numero
- Scrivi un programma che esegue la somma di tutti i numeri multipli di 5 compresi tra 10 e 100.
- Scrivi un programma che effettua il conto alla rovescia a partire da un valore minore di 20 inserito dall'utente (accertarsi che il numero inserito dall'utente sia minore di 20).
- Scrivi un programma che effettua il lancio di due dadi e visualizza i risultati.
- Scrivi un programma che genera NUM numeri random (NUM inserito dall'utente) e visualizza quanti numeri pari e quanti numeri dispari sono stati generati.
- Scrivi un programma che effettua la somma dei numeri inseriti dall'utente fino

- a raggiungere il numero 1000 e indica quanti numeri sono stati sommati.
- Scrivere un programma che permetta di fare una partita a dadi
 - tra due giocatori. Il gioco è fatto così. Un giocatore lancia due dadi e i risultati vengono memorizzati. Lo stesso fa l'altro giocatore. La partita può finire 2-0,1-0,0-0, 0-1, 0-2 a seconda di come si combinano i dadi. I dadi si combinano in questo modo: il maggiore con il maggiore e il minore con il minore. Per esempio se lancio Luca : 5 3, lancio Paolo: 5 3 il risultato è 0-0 cioè parità; se lancio Luca : 5 3, lancio Paolo: 3 2 il risultato è 2-0 cioè vince Luca e così via. Il programma all'inizio chiederà il nome dei giocatori e poi dopo aver mostrato il lancio dei due dadi mostrerà il risultato della partita in modo opportuno.
 - Realizzare un algoritmo che data in input una sequenza di numeri chiusa dallo zero determini la loro somma e la media. Nel caso l'utente non digiti lo zero non sarà possibile inserire più di 10 numeri
 - Scrivere un algoritmo che presi in input dei numeri calcoli la somma dei loro valori assoluti. Il ciclo termina su richiesta dell'utente
 - Scrivere un algoritmo che presi in input N numeri calcoli la somma dei soli numeri dispari
 - Il prezzo del biglietto del cinema è di 8 euro. Le persone di età minore di 5 anni e maggiore di 75 non pagano, mentre le persone con età compresa tra 5 e 10 anni hanno uno sconto del 40%. Le altre pagano normalmente
 - Scrivi un programma che legge un numero NUM da tastiera e quindi successivamente esegue la somma di NUM numeri inseriti dall'utente.
 - Scrivere un algoritmo che presi in input una sequenza di numeri con uscita su richiesta dell'utente ed un numero X preso in input calcoli: quanti numeri sono maggiori di X, quanti sono minori di X e quanti sono uguali a X
 - Esercizi con switch case: dato il numero di un mese stampare il mese in lettere corrispondente
 - Scrivere un programma che prende in input il numero dei lati compreso tra 3 e 8 e stampa i nomi dei poligoni regolari con tale numero di lati
 - Leggi il valore di un numero compreso tra 0 e 2 che corrispondono agli stati di un semaforo e visualizza una parola che lo traduce nel colore corrispondente.
 - Scrivere un algoritmo che visualizza i primi 100 numeri pari
 - Generalizza l'esercizio precedente facendolo operare su N numeri
 - Scrivere un algoritmo che visualizza i primi 100 numeri dispari
 - Costruire gli algoritmi che generano e stampano le seguenti sequenze
 - 1, 3, 5, 7, 9, ...,99
 - 1, 4, 9, 25, ...,2500
 - 1, -2, 3, -4, 5, -6,...99, -100
 - $\frac{1}{2}$, $\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$, ..., $\frac{99}{100}$
 - Prese in input il nome della città e la sue temperature massima e minima per

una serie di città italiane, stabilire la città con la temperatura massima più alta e quella con la temperatura minima più bassa e mostrare in output questi dati (nomi delle città max e min e loro temperature). L'immissione dei dati terminerà quando l'utente lo Desidera.

- Chiedere il nome ad una persona e salutarlo per 10 volte.
- Chiedere il nome ad una persona e salutarlo per 10 volte inserendo anche il numero tipo "saluto n°1 ciao" per 10 volte
- Chiedere all'utente quante volte vuole essere salutato e quindi salutarlo per le volte scelte
- Stampare tutti i numeri pari tra 1 e 10
- Scrivere un algoritmo che calcoli la somma dei quadrati dei primi K numeri naturali successivi a un numero naturale N, noti K ed N.
- Scrivere un algoritmo che dato un numero intero n positivo determini tutti i suoi divisori interi.
- Fattoriale di un numero: Scrivere un programma che dato un numero stampi il fattoriale del numero. Ricordo che dato n il fattoriale di n (indicato in matematica con $n!$) è dato da $n*(n-1)*(n-2)*...*2*1$. Quindi per esempio dato 5, $5!$ vale $5*4*3*2*1$
- Scrivere un programma che dato un numero stampi i fattoriali di tutti i numeri tra 0 e il numero inserito. Ricordo che $0! = 1$ e che $1! = 1$.
- Scrivere un programma C che dati in input N numeri determini il fattoriale di ognuno. Dato il numero n il fattoriale di n, indicato con il simbolo $n!$, è dato da $n * (n-1) * (n-2) * ... * 2 * 1$. Per esempio il fattoriale di 4, cioè $4!$, è pari a $4! = 4*3*2*1 = 24$
- Dato un numero intero maggiore di 0 verificare se è un numero primo. Un numero è primo se è divisibile solo per 1 e per se stesso e non è divisibile per nessun altro numero
- Dato un numero intero n maggiore di 2 stampare tutti e soli i numeri da 3 a n che sono numeri primi. Quindi se per esempio il numero inserito è 15 stamperò: 3, 5, 7, 11, 13 e se inserisco 22 stamperò: 3, 5, 7, 11, 13, 17, 19
- Calcolare il prodotto tra due numeri interi positivi con solo somme
- Calcolare a elevato b usando solo con solo somme.
- Scrivere un programma permetta di mantenere una specie di borsellino personale. Il borsellino funziona così: quando vi arrivano dei soldi perchè ve li hanno regalati o sono la vostra paghetta settimanale etc. aggiungete i soldi al vostro saldo attuale. Quando spendete dei soldi togliete i soldi spesi dal vostro saldo attuale. Il programma dovrà presentare un menù con le varie operazioni che potete fare che saranno la registrazione di un'entrata o una uscita e poi permettere la visualizzazione del saldo attuale e delle spese totali e delle entrate totali oppure uscire dal programma. State attenti che non potete spendere più di quello che avete nel saldo.
- Esercizio: Scrivere un programma che prenda in input il numero del mese e il

numero del giorno iniziale del mese con Lunedì = 1, Martedì = 2, Mercoledì = 3, ..., Domenica = 7. Il programma dovrà stampare il calendario nella forma mostrata nel seguente esempio: Se per esempio il mese è Agosto, quindi 8, ed il primo del mese è Giovedì quindi 4 allora verrà stampato

<u>Lun</u>	<u>Mart</u>	<u>Merc</u>	<u>Giov</u>	<u>Ven</u>	<u>Sab</u>	<u>Dom</u>
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

- Scrivi un programma che visualizzi i primi 100 numeri dispari a gruppi di 5 (andando quindi a capo ogni 5 numeri). Dopo aver scritto il programma (anche se sbagliato) eseguire basandosi su di esso la tabella di traccia nel quaderno (per tutti gli alunni).
- Stampare tutti i numeri compresi tra 1 e N accertandosi con un ciclo a controllo in coda che N sia maggiore o uguale a 100 andando a capo ogni 10 numeri stampati
- Dato un numero intero positivo (accertandosi che lo sia) stabilire e comunicare se è primo o meno
- Preso un numero intero positivo in input (accertandosi che lo sia) stampare tutti i numeri primi compresi tra 1 e il numero intero positivo stesso
- Stampa di una riga della tabellina del 10 di un numero inserito da tastiera. Upgrade: stampa di tutte le tabelline fino al numero inserito.
- Stampa di tutte le tabelline con la cornice
- Determinare il quadrato di un numero intero usando la regola che dice che il quadrato di un numero intero n è dato dalla somma dei primi n numeri dispari
- Determinare il quadrato di tutti i numeri interi compresi tra 1 ed n (intero positivo), attraverso la regola: $5^2 = 1+3+5+7+9=25$, cioè il quadrato del numero **5** è dato dalla somma dei primi **5** numeri dispari)
- Scrivi un programma che calcola quante volte il numero 7 compare nei numeri compresi tra 1 e 100.
- Scrivi un programma che disegna sullo schermo 3 cornici quadrate concentriche costituite dal carattere "*" rispettivamente di lato 12, 8, e 4.
- Scrivi un programma per la rappresentazione di un triangolo così definito: si

riempiono le righe del triangolo con numeri consecutivi e partendo da 1 nell'angolo in alto a sinistra e proseguendo sino a n con $n \leq 9$. Si prenda n in input accertandosi che sia minore di 10. Per esempio se si prende 4 il triangolo sarà

```
1
12
123
1234
```

- **UN PROBLEMA DI COMBINAZIONI:** In un quiz vengono poste a un concorrente un certo numero di domande facili e difficili:
 - se risponde giusto a una domanda facile guadagna 2 punti, a una domanda difficile 4 punti;
 - se risponde sbagliato a una domanda facile perde 4 punti, a una domanda difficile perde 2 punti.
 Dopo 20 domande il concorrente ha totalizzato 30 punti: quali possibili risposte ha dato suddivise tra facili e difficili e giuste ed errate? Indicare almeno tre possibili combinazioni.
- Scrivi un programma che effettui il controllo sul peso di confezioni di pasta impacchettate manualmente da un gruppo di operatori. Inserendo la percentuale di tolleranza e il valore nominale della confezione, il programma deve segnalare se il pacchetto rientra nella norma oppure deve essere scartato. Dopo aver individuato 5 pacchetti “fuori norma”, il programma dovrà terminare indicando la percentuale di errori di impacchettamento.
- Diagramma di flusso dell'algoritmo per il calcolo di b elevato e con b ed e interi ed e maggiore o uguale a 0. Implementano quindi l'algoritmo in linguaggio C.
- Calcolare b elevato e, con e qualsiasi (anche negativo)
- Scrivere un programma che usando le funzioni calcoli il minimo tra due numeri. Scrivere 4 versioni del programma nel quaderno.
- Scrivere un programma che utilizzando un sottoprogramma stampi una riga di un dato carattere preso in input
- Scrivete utilizzando le funzioni stampaRiga dei programmi che stampano dei triangoli isosceli e scaleni così fatti. La lunghezza del lato e il carattere lo prendete in input

```
*           *           * * *           * * *
* *         * *         * *         * *
* * *       * * *       *           *
```

- quadrati e cubi: scrivere un programma che prende in input un numero intero N positivo accertandosi che lo sia e stampi i quadrati dei numeri da 1 N e i

cubi dei numeri da uno a N. Il programma deve usare tre funzioni: controllo input, quadrato e cubo

- fattoriale: scrivere un programma che calcoli il fattoriale dei primi N numeri con N preso in input e controllando che sia maggiore di zero. Usare funzione di controllo input e funzione fattoriale
- lancio Dadi: lanciare per 100.000 volte un dado a 6 facce e calcolare il numero di uscite per ciascuna faccia e la percentuale di uscita di ciascuna faccia
- Per ogni esercizio successivo svolgere i seguenti passi scrivendoli (domande e risposte)
 - che cosa deve restituire la funzione? → determina il tipo di dato restituito dalla funzione
 - Quali dati deve conoscere la funzione per poter svolgere il suo lavoro? Quale tipo hanno tali dati? Questi dati si tramuteranno nei parametri formali o argomenti della funzione
 - codifica della funzione
 - utilizzo della funzione in un programma come specificato
 - Scrivere una funzione che stabilisca se un numero è pari o dispari
 - utilizzando la funzione precedente scrivere un programma che prende un numero in input e comunicare se è pari o dispari
 - utilizzando la funzione precedente scrivere un programma che prenda una serie di numeri in input chiusi con il numero zero e comunichi per ciascuno di essi se è pari o dispari
 - Scrivere una funzione che preso un numero intero positivo stabilisca se è un numero primo o meno. Suggerimento: un numero è primo se è divisibile solo per 1 e per se stesso e se non è divisibile per nessun altro numero. Pertanto si può ipotizzare inizialmente con una variabile FLAG (bandiera) che esso sia primo e poi si cicla per tutti i numeri necessari e se si trova un divisore si imposta la variabile FLAG a false e si esce dal ciclo. Alla fine si controlla il valore della variabile FLAG e in base ad essa si saprà il risultato. Un nome adatto per tale variabile è isPrimo
 - scrivere un programma che utilizzando la funzione precedente prende un numero in input e comunica se è primo o meno
 - utilizzando la funzione precedente prendere una serie di numeri in input, sino a che l'utente lo desidera, e comunicare per ciascuno di essi se è primo o meno.
- Scrivere una procedura leggiDiversi che si assicura di prendere in input 3 numeri diversi fra loro ed usarla per acquisire tre numeri e li stampi.
- Scrivere un programma che dati due numeri li ordini in senso crescente. Per

- farlo utilizzi una procedura con due parametri per riferimento.
- Scrivere un programma che ordini tre numeri usando la procedura `ordina2Numeri()` dell'esercizio precedente. Il programma userà una procedura `ordina3Numeri` che richiamerà al suo interno la procedura `ordina2Numeri()`
 - Scrivere una funzione con questa firma `int menu()` che proponga all'utente un menu con le seguenti voci
Cosa vuoi fare?:
 1 → calcolare area rettangolo
 2 → calcolare area triangolo
 3 → calcolare area cerchio
 4 → calcolare perimetro rettangolo
 5 → calcolare misura circonferenza
 6 → uscire
 - a) la funzione deve restituire un intero che costituisce la scelta dell'utente
 - b) Scrivere una funzione che calcoli l'area di un rettangolo
 - c) Scrivere una funzione che calcoli l'area di un triangolo
 - d) Scrivere una funzione che calcoli l'area di un cerchio
 - e) Scrivere una funzione che calcoli il perimetro di un rettangolo
 - f) Scrivere una funzione che calcoli la misura di una circonferenza
 - g) Scrivere un programma che usando le funzioni di cui ai punti a, b, c, d, e, f e g chieda ripetutamente all'utente cosa vuole fare presentandogli il menù e poi, dopo aver chiesto gli input opportuni per ciascun caso, calcoli quanto proposto dal menu usando le funzioni precedentemente definite.
 - scrivere un programma che, usando un solo sottoprogramma, calcoli l'area e il perimetro di un rettangolo.
 - scrivere un programma che usando una funzione incrementa di 1 una variabile. La funzione si chiama `incr`
 - scrivere un programma che usando una procedura incrementa di 1 una variabile. La procedura si chiama `incr`
 - scrivere un programma che usando una funzione incrementa di una certa quantità `q` una variabile. La funzione si chiama `incrQ`
 - scrivere un programma che usando una procedura incrementa di di una certa quantità `q` una variabile. La procedura si chiama `incrQ`
 - Risolvi gli esercizi seguenti definendo una funzione per effettuare le eventuali operazioni di lettura e una seconda funzione che effettua l'elaborazione. Il `main()` deve contenere un ciclo che permetta di ripetere l'esecuzione del programma che termina quando l'utente digita un carattere particolare.
 - Scrivi un programma che acquisisce il valore `N` intero positivo e maggiore o uguale a due e presenta un menu per scegliere una delle seguenti opzioni:
 1. visualizza una riga di `N` asterischi;

2. visualizza una colonna di N asterischi;
3. visualizza una diagonale di N asterischi;
4. visualizza un quadrato con il lato di N asterischi;
5. visualizza un triangolo rettangolo con i due cateti di N asterischi;
0. Fine.

- Simulare il gioco del lotto di una città con l'uso della funzione random e array
- Scrivere un programma che permetta di estrarre 5 numeri diversi fra loro e li inserisca in un vettore (soluzione su classroom)
- Scrivere una funzione C/C++ che, a partire da un vettore numerico di dimensione n fornito come parametro, copi in un secondo vettore (anch'esso fornito come parametro) i soli elementi del primo vettore compresi tra un valore minimo a e un valore massimo b (anch'essi argomenti della funzione); la funzione deve restituire il numero di elementi inseriti nel secondo vettore.
- Scrivere un programma C che, utilizzando la funzione precedente, visualizzi gli elementi del secondo vettore dopo aver inserito da programma (per esempio con rand()) i valori del primo vettore.
- Scrivi un programma che legge un numero positivo in ingresso inferiore a 10.000 e calcola i suoi divisori memorizzandoli in un vettore. Quindi li visualizza sullo schermo al termine dell'elaborazione.
- Genera casualmente 10 numeri e memorizzali in un array. Stampa l'array. Scambia il massimo e minimo elemento e quindi visualizza nuovamente l'array (supponendo che i valori dell'array siano tutti distinti tra loro).
- Scrivi un programma che genera casualmente un numero compreso tra 0 e 255 e lo trasforma in un numero binario utilizzando un vettore di dimensione 8.
- Scrivere un programma che permetta di acquisire il nome e cognome di ciascun alunno e il voto scritto, orale e pratico in una materia anch'essa presa in input. Si vuole produrre in uscita una tabella che riproduca i dati in input con in più la media dei voti di ciascun alunno
- Scrivi un è programma che genera casualmente 30 numeri e li memorizza in due vettori: il primo vettore deve contenere solo i numeri pari mentre il secondo vettore i numeri dispari.
- Leggi 10 numeri interi da tastiera memorizzali in un array e visualizzali in sequenza senza stampare uno stesso numero due volte. Per esempio:
 - valori dell'array: 15, 3, 5, 3, 11, 5, 15, 5, 15, 11;
 - valori stampati: 15, 3, 5, 11.
- Dato un elenco di MAX_ALUNNI studenti di una classe con il nome e il numero di risposte esatte (E), quelle sbagliate (S) e quelle non date (N) di una verifica, calcolare il punteggio di ciascuno secondo la seguente formula: $P = E - (1/2) * S - (1/4) * N$
 Alla fine si mostri una tabella dei risultati di questo tipo (i dati sono solo un esempio)

Nome	E	S	N	P
Luca Pinna	7	2	1	5.75
Nino Pes	8	2	0	7
Eva Pisu	9	0	1	8.75

Verifica migliore

Eva Pisu 9 0 1 8.75

Si utilizzi obbligatoriamente

1. un sottoprogramma con parametri che permetta di caricare il numero di risposte esatte (E), quelle sbagliate (S) e quelle non date (N) di una verifica per ciascun alunno precedentemente caricato. Il sottoprogramma deve mostrare il nome dell'alunno e chiedere

l'inserimento dei dati.

2. un sottoprogramma con parametri per calcolare i punteggi di ciascuno studente e memorizzarli

3. un sottoprogramma con parametri che restituisce l'indice dell'alunno con prova migliore. In caso vi sia più di un alunno con prova migliore si restituisca solo il primo indice trovato

4. un sottoprogramma con parametri per stampare la tabella finale e la verifica migliore

Suggerimenti: `strlen(v)`; //restituisce la lunghezza della stringa v

- Leggi N numeri, memorizzali in un vettore e ribalta il vettore. Per esempio:
– valori dell'array: 15, 23, 35, 46, 51, 68, 12, 72;
– valori ribaltati: 72, 12, 68, 51, 46, 35, 23, 15.
- Scrivi un programma che generi casualmente 30 numeri di valore minore di 1000, li memorizza in un vettore e visualizzi dal più piccolo al più grande i numeri pari
- Esercizi Matrici: definire una matrice 3 x 4 e inizializzarla a 0 e stamparla
inserire nella posizione (seconda riga, prima colonna) il valore 25 e stampare la matrice
riempire la seconda riga con il valore 36 e stampare la matrice
riempire la terza colonna con il valore 10 e stampare la matrice
Ridefinire la matrice come una matrice quadrata 4x4 e stampare la matrice
riempire la diagonale principale con il valore 12 e stampare la matrice
riempire la diagonale principale con un valore casuale tra 1 e 90 e stamparla
riempire la diagonale secondaria con il valore 17 e stampare la matrice
riempire la diagonale secondaria con un valore casuale tra 1 e 90 e stamparla
riempire la matrice triangolare superiore (tutte le caselle sopra la diagonale principale e la diagonale stessa) con valori casuali tra 1 e 90
riempire la matrice triangolare inferiore (tutte le caselle sotto la diagonale

- principale e la diagonale stessa) con valori casuali tra 1 e 90
- Una stazione metereologica esegue ogni giorno ad intervalli di un'ora un insieme di letture per le ore più calde (11:00-16:00). Si vuole sapere, relativamente alla settimana di ferragosto, in quale giorno e ora si è registrata la temperatura massima e quale è stata la temperatura media per ogni fascia oraria
 - L'utente inserisce una matrice $N \times M$ composta da numeri interi. Il programma scambia le righe pari con quelle dispari.
 - Supponi di avere in memoria una matrice rettangolare $M \times N$ di numeri interi indicanti l'altezza di alcune persone. Qual è la persona più alta: la più bassa tra le più alte di ogni riga, oppure la più alta tra le più basse di ogni colonna? Scrivi un programma che verifichi quale affermazione è vera
 - Realizza un programma che, partendo da un array che contiene i dati personali dei calciatori di una squadra di calcio, permetta di calcolare:
 - il totale dei goal segnati dalla squadra, specificando:
 - i goal segnati dagli attaccanti (numero di maglia da 7 a 11);
 - i goal segnati dai centrocampisti (numero di maglia da 4 a 6);
 - i goal segnati dai difensori (numero di maglia da 1 a 3);
 - il totale dei goal subiti dalla squadra.
 Il programma inoltre deve visualizzare i dati personali di tutti i giocatori e per ognuno di essi memorizzare:
 - il numero di maglia;
 - il numero di goal segnati (subiti, se è un portiere).
 Pensate alle possibili strutture dati che vi servono per risolvere questo problema tra vettori, matrici, vettori paralleli o corrispondenze tra vettori e matrici
 - Definisci la struttura dati per effettuare le prenotazioni dei biglietti di uno stadio dove tutti i posti sono numerati con un codice parlante del tipo AZSN123 (A = anello, Z = zona, S = settore, N = orientamento 123 = posto).
 - 2 Progetta un programma di gestione delle schedine del Totocalcio che, partendo da un array di dimensione prefissata che contiene in ogni cella il codice della giocata, la data della giocata e la colonna con i tredici segni, effettui la copia in un altro vettore solo delle giocate relative all'anno e al mese richiesto dall'utente.
 - Implementa un programma che, dato un vettore di studenti (il tipo `Studente` conterrà nome, cognome e i voti scritto, orale e pratico di ogni quadrimestre in una determinata materia), individui e restituisca il nome dello studente più bravo con la sua media cioè quello con media generale più alta.
 - Stampare inoltre l'elenco degli studenti con i loro voti
 - Implementa un programma che, dato un vettore di studenti (il tipo `Studente` conterrà nome, cognome e i voti scritto, orale e pratico di ogni quadrimestre), individui e ritorni il nome dello studente più bravo nelle diverse materie e con

- media generale più alta.
- Definire la struttura dati Auto che descrive un'automobile in base a marca, cilindrata, anno di immatricolazione e acquirente. L'acquirente è caratterizzato dai soli nome e cognome. Scrivere un programma C che, dichiarata la tabella Autosalone di tipo Auto (per comodità se ne fissi la dimensione a 10 elementi), consenta di inserire da tastiera i dati delle auto vendute e di visualizzare il solo cognome degli acquirenti di auto di cilindrata superiore a 1500 cc, oltre al numero totale di auto che sono state immatricolate in un anno richiesto all'utente.

I Docenti **Alberto Saba** e **Daniela Floris**