I I S *"Ettore Majorana"*
Via A. De Gasperi, 6 - 20811 Cesano Maderno (MB)

COMPITI ESTIVI

CLASSE	3IA2	DISCIPLINA	INFORMATICA
DOCENTE	Prof. TRAMONI RICCARDO	A.S.	2025/2026

ESERCIZIO 1 (OBBLIGATORIO)

Una PILA in Informatica è una struttura di dati dinamica che segue la logica LIFO (Last In First Out), per cui l'ultimo elemento inserito nella struttura (Last In) è l'unico ad essere visibile e il primo a poter essere tolto (First Out).

Nel linguaggio C una pila può essere implementata con la seguente struttura:

```
typedef struct {
    int capacity; //capacity e' la grandezza (iniziale) della pila
    int head; //head e' un indice intero che segna la posizione dell'ultimo elemento inserito (testa della pila)
    int *buffer; //buffer e' l'array dinamico che contiene i dati (in questo caso numeri interi)
} pila;
```

Di default, quando una pila è vuota, head = -1; invece quando è piena head = capacity-1. Solitamente, nonostante il buffer sia un array dinamico, capacity non viene modificata, e la pila si dice piena quando la capacità è raggiunta.

Una pila di numeri interi può essere inizializzata con la seguente funzione:

```
pila *createPila(int capacity){ //l'argomento e' la capacita' della pila
    pila *p=(pila*)malloc(sizeof(pila)); //la memoria della pila e' allocata dinamicamente, con un cast a pila*
    p->buffer = malloc(sizeof(int)*capacity); //il buffer e' l'array dinamico di dimensione capacity
    p->capacity = capacity; //come visto in classe, si usa -> per non scrivere (*p).
    p->head = -1; //la pila viene creata vuota
    return p; //la funzione e' di tipo pila*, quindi ritorna la pila stessa (o meglio, il puntatore alla pila)
}
```

(Provare a) scrivere un'implementazione per le seguenti funzioni che operano su una pila:

- bool isEmpty(pila * p): ritorna TRUE se la pila è vuota (verifica se head=-1);
- bool isFull(pila * p): ritorna TRUE se la pila è piena, cioè se la testa corrisponde con l'elemento di indice capacity-1;
- bool push(int valore, pila * p): ritorna FALSE se la pila è piena, altrimenti inserisce 'valore' in testa alla pila e ritorna TRUE;
- int top(pila * p): ritorna l'elemento in testa alla pila, se non è vuota;
- int pop(pila * p): come top(), ma rimuove l'elemento in testa.

Poi, testare la correttezza delle funzioni in un programma.

ESERCIZIO 2 (FACOLTATIVO)

Nel linguaggio Python, un dizionario è un insieme di coppie chiave:valore, in cui le chiavi possono essere qualsiasi tipo di variabile e devono essere tutte diverse tra di loro.

Per esempio: d = {'a': 1, 'b': 2, 'c': 3} ha tre elementi interi associati a tre chiavi di tipo carattere.

- il comando d[‘a’] restituisce il valore associato alla chiave ‘a’;
- il comando d[‘a’] = 42 modifica il valore associato a una chiave già esistente;
- il comando d[‘d’] = 4 crea un nuovo elemento con chiave ‘d’ e valore 4;
- il comando del d[‘c’] elimina l’elemento (chiave e valore) associato alla chiave ‘c’.

Un dizionario supporta diversi metodi (funzioni), tra cui:

- d.items(): restituisce gli elementi di d come un insieme di tuple;
- d.keys(): restituisce la lista delle chiavi di d;
- d.values(): Restituisce la lista dei valori di d;
- d.copy(): crea e restituisce una copia di d;
- d.update(d2): aggiunge gli elementi del dizionario d2 a quelli di d;
- d.clear(): rimuove tutti gli elementi di d.

Per scorrere gli elementi di un dizionario si può utilizzare un ciclo: for chiave in d.keys(): .

Nel linguaggio Python, una matrice può essere definita, in modo tanto scomodo quanto divertente, come un dizionario in cui le chiavi sono delle tuple corrispondenti alle coordinate di un elemento della matrice, e il valore è il valore stesso dell’elemento della matrice.

Un esempio è il seguente:

matrix = {(0,0):42, (0,1):7, (1,0):66, (1,1):0}

La matrice matrix ha due righe e due colonne: l’elemento in posizione matrix[0][0] vale 42, e così via.

Scrivere nel linguaggio Python la funzione somma(m1, m2, n) che riceve in argomento due matrici quadrate di uguale dimensione n, e ritorna una matrice che contiene la somma tra le due matrici in argomento. Fare la stessa cosa con la funzione prodotto(m1, m2, n).

ESERCIZIO 3 (NECESSARIO)

Guardare un film, scelto dalla lista condivisa su Classroom, **oppure** un qualsiasi film a scelta che non abbiate mai visto prima. Possibilmente, in lingua inglese (se è la lingua originale del film).

Alcune accortezze:

- i compiti **NON** saranno valutati;
- i compiti **NON** sono assegnati con lo scopo di rovinarvi le vacanze, quindi se non volete farli o li fareste fare a ChatGPT (o un altro coso), piuttosto **NON** fateli;
- l’esercizio 1 ha lo scopo di cominciare a farvi prendere dimestichezza col primo argomento di quarta, che riguarda le strutture dinamiche di memoria come le pile e le code nel linguaggio C.

BUONE VACANZE!

Cesano Maderno, 17/06/2026 Prof. Riccardo Tramoni