

## Istruzioni

- Tempo disponibile: 90 minuti.
- Non è permesso l'uso di dispositivi elettronici (a parte il PC della propria postazione).
- E' consentito l'uso di materiale didattico cartaceo. Non è consentito l'uso di materiale didattico elettronico.
- Visual Studio Code è installato su tutte le postazioni. Si può comunque usare l'editor che si preferisce fra quelli installati.

## Esercizio 1 (max. 9 punti)

L'algoritmo in figura 1 riceve in input due numeri interi non negativi  $b$  ed  $e$  e produce il valore della potenza  $b^e$ . Nella figura, IN significa input, OUT output,  $\leftarrow$  assegnamento e tutte le operazioni sono in aritmetica intera.

Scrivere una **procedura** di nome **potenza**, con parametri e tipo di ritorno appropriati, che implementi fedelmente<sup>1</sup> l'algoritmo in figura 1 senza usare l'istruzione **goto**.

Utilizzare la procedura **potenza** in un programma che

1. richiedi all'utente due numeri interi positivi  $M$  e  $N$ ;

2. stampi  $\sum_{k=0}^N M^k$ , cioè

$$M^0 + M^1 + \dots + M^{N-1} + M^N.$$

Ad esempio, se l'input fosse

3 10, un output corretto sarebbe il seguente:

88573

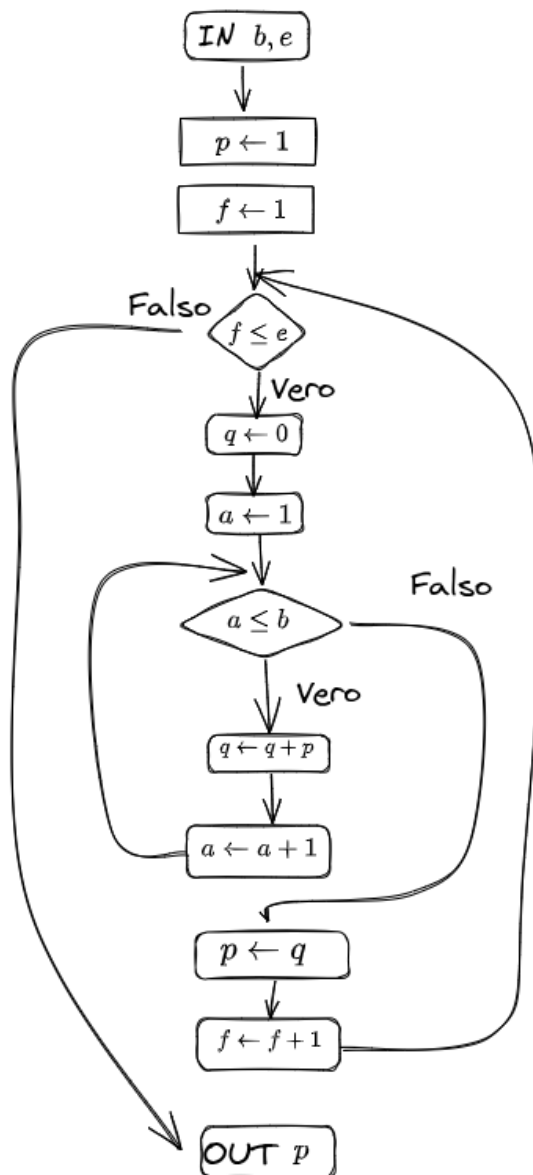


Figura 1: Potenza

Per consegnare, caricare un file di nome **Esercizio1.c** con tutto il codice richiesto. L'elaborato sarà valutato per

- Identificazione delle strutture dati e degli algoritmi appropriati alle specifiche

<sup>1</sup>L'implementazione di altri algoritmi non è una soluzione valida.

- Corretta implementazione di strutture dati e algoritmi
- Strutturazione del programma in funzioni come richiesto dalle istruzioni
- Stile (chiarezza, utilizzo di costrutti appropriati)

I programmi non compilabili saranno valutati 0 punti. L'utilizzo di costrutti estranei al programma della prova sarà penalizzato.

## Esercizio 2 (max. 2 punti)

Nel programma allegato [d2.c](#), riportato in figura 2, qual è il numero massimo di record di attivazione (frame) allocati sullo stack se si dà come input il proprio numero di matricola?

Per rispondere, caricare un file di nome [Esercizio2.txt](#) contenente solo il valore richiesto.

Valutazione:

- 2 punti per risposta corretta;
- 0 punti per risposta non data (file [Esercizio2.txt](#) non caricato)
- -1 punto per risposta errata.

```
#include <stdio.h>

int f(int n) {
    if (n < 10000000)
        return f(n * 3);
    else
        return 0;
}

int main(void) {
    int M;
    scanf("%d", &M);
    M = f(M);
    return M;
}
```

Figura 2: Codice a cui si riferisce l'esercizio 2.