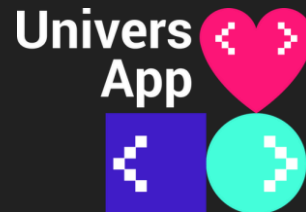


INTRODUZIONE ALLA PROGRAMMAZIONE IN LINGUAGGIO C



Array monodimensionali: Computational Thinking

Informaticage e UniversApp!



Somma in range

Dato un Array **V** di **N** elementi di tipo **numerico (int, float, double, long..)**.
Calcolare la somma $S(i, j)$ di un intervallo chiuso $[i, j]$ di **V**.

$$S(i, j) = \sum_{k=i}^j v[k]$$

Array di esempio:

6	6	4	8	-10	3	2	5	-3
---	---	---	---	-----	---	---	---	----

Input:

$i = 4$

$j = 7$

Output:

$$S(i, j) = S(4, 7) = v[4] + v[5] + v[6] + v[7] = -10 + 3 + 2 + 5 = 0$$

Prefix sum array

Dato un Array **V** di **N** elementi di tipo **numerico (int, float, double, long..)**..
Trasformare l'Array **V** in un array di somme prefisse basate su **V**.

Si definisce **P_{refix} S_{um} A_{rray}** (array delle somme prefisse)
un **Array V** tale che per ogni elemento $V[i]$ di **V**, $V[i] = S(0, i) = \sum_{k=0}^i v[k]$

Array **V** di esempio:

6	6	4	8	-10	3	2	5	-3
---	---	---	---	-----	---	---	---	----

PSA(V):

6	12	16	24	14	17	19	24	21
---	----	----	----	----	----	----	----	----

↑ ↑ ↑ ↑
6 6+6 6+6+4 6+6+4+8 ...

Inversione in loco

Dato un Array **V** di **N** elementi di tipo **T**.
Invertire **V** in loco (Senza utilizzare un secondo array).

V before:

6	6	4	8	10	3
---	---	---	---	----	---

V after:

3	10	8	4	6	6
---	----	---	---	---	---

Duplicati

Dato un Array **V** di **N** elementi di tipo **T**.
Identificare e stampare in output i duplicati:

Array **V**:

6	6	4	8	-10	3	2	8	3
---	---	---	---	-----	---	---	---	---

Output:

Duplicati: 6, 8, 3