

## Esercizio 1

Creare un file “main.c”. Nel file, si completi la seguente funzione utilizzando comandi del linguaggio C:

```
double distanza (double ax, double ay, double bx, double by) {  
    ...  
}
```

La funzione deve restituire la distanza euclidea tra i due punti a e b. Si utilizzi la formula:

$$d(a, b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

La radice si calcola con la funzione vista a lezione.

## Esercizio 2

Creare un file “main.c”. Nel file, si completi la seguente funzione utilizzando comandi del linguaggio C:

```
uint32_t x_alla_x (uint32_t x) {  
    ...  
}
```

La funzione deve restituire il risultato della formula:  $x^x$ .

Se  $x = 0$  la funzione deve ritornare 0.

**Domanda:** qual è il valore massimo di  $x$  per cui la funzione calcola correttamente il risultato?

## Esercizio 3

Creare un file “main.c”. Nel file, si completi la seguente funzione utilizzando comandi del linguaggio C:

```
void x_alla_x (uint32_t *x) {  
    ...  
}
```

La funzione deve modificare il valore a cui punta  $x$ , sostituendovi il risultato della formula:  $x^x$

## Esercizio 4

Creare un file “main.c”. Nel file, si completi la seguente funzione utilizzando comandi del linguaggio C:

```
int calcola_minimo (int *v, size_t n) {  
    ...  
}
```

La funzione deve restituire il minore tra i numeri interi del vettore  $v$ , che contiene  $n$  elementi. Il vettore  $v$  conterrà sempre almeno un elemento.

## Esercizio 5

Creare un file “main.c”. Nel file, si completi la seguente funzione utilizzando comandi del linguaggio C:

```
int somma_sottrai (int *v, size_t n) {  
    ...  
}
```

La funzione `somma_sottrai` calcola il valore da ritornare sommando gli elementi con indice pari e sottraendo quelli con indice dispari.

Esempio:

$v$ : 

1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---

$risultato = 1 - 2 + 3 - 4 + 5 - 6 + 7 - 8 + 9$

Il vettore  $v$  conterrà sempre almeno un elemento.

## Esercizio 6

Creare un file “main.c”. Nel file, si completi la seguente funzione utilizzando comandi del linguaggio C:

```
float *ripeti (float val, uint32_t n) {  
    ...  
}
```

La funzione deve allocare dinamicamente nella memoria heap un nuovo vettore di numeri in virgola mobile a 32 bit, che contenga il valore  $val$  ripetuto  $n$  volte. La funzione ritorna un puntatore al vettore così creato.

Qualora  $n$  sia 0, la funzione restituisce NULL.