

---

## APPELLO DI PROGRAMMAZIONE I

14 GIUGNO 2021

---

### INDICAZIONI GENERALI

- Scaricare il file di ogni esercizio e riconsegnarlo senza modificarne il nome.
  - I file non consegnati o consegnati con errori di compilazione non verranno presi in considerazione.
  - Si possono utilizzare funzioni aggiuntive non presenti nei file modello e aggiungere linee di commento alle funzioni già implementate nel modello.
  - I file possono essere consegnati più volte. Per ogni esercizio, solo l'ultimo file consegnato sarà considerato valido.
  - I warning ottenuti compilando con l'opzione `-Wall` avranno un peso negativo sul voto finale.
  - Si consiglia di utilizzare il comando `ulimit -v 500000` per limitare l'utilizzo delle risorse al terminale su cui viene eseguito il comando ed evitare spiacevoli inconvenienti dovuti ad eccessive allocazioni di memoria.
- 

### COMPITO A

---

#### Esercizio 1 [11 punti] File `ESA_14062021_A_1.c`

Si completi il file `ESA_14062021_A_1.c` di modo che il sottoprogramma `calcola`, ricevuta in ingresso una matrice di valori interi di dimensione  $N \times N$  (con  $N$  pari a 5) calcola e restituisce l'indice della colonna in cui la massima differenza (in valore assoluto) ottenuta tra due suoi elementi, sia massima. Nel caso ci sia più di una colonna con uguale differenza massima, restituire l'indice della prima colonna trovata. Ad esempio, se il

sottoprogramma riceve la seguente matrice  $\begin{bmatrix} 2 & 5 & 10 & 15 & 18 \\ 11 & 7 & 7 & \mathbf{20} & 15 \\ 5 & 5 & 7 & 11 & 22 \\ 6 & 1 & 3 & 16 & 19 \\ 7 & 1 & 2 & \mathbf{10} & 18 \end{bmatrix}$  restituirà come indice 3: tale colonna ha

come massima differenza 10 (differenza tra 20 e 10), ed è maggiore della differenza massima delle altre quattro colonne, rispettivamente 9, 6, 8 e 7.

### Esercizio 2 [11 punti] File `ESA_14062021_A_2.c`

Si completino **tutte le funzioni dichiarate ma non definite** nel programma.  
In particolare, la seguente funzione `main()` fornita nel file stub:

```
int main() {
    struct letter_t *phrase[LENGTH] = {NULL};
    put_word(phrase, 0, LENGTH, "hello");
    put_word(phrase, 1, LENGTH, "darkness");
    put_word(phrase, 2, LENGTH, "my");
    put_word(phrase, 3, LENGTH, "old");
    put_word(phrase, 4, LENGTH, "friend");
    print_phrase(phrase, LENGTH, ' ');
    printf("\n");

    free_word(phrase[0]);
    free_word(phrase[1]);
    free_word(phrase[2]);
    free_word(phrase[3]);
    return 0;
}
```

deve fornire il seguente output:

```
hello darkness my old
```

### Esercizio 3 [11 punti] File `ESA_14062021_A_3.c`, `ESA_14062021_A_3.pdf`

Completare il file `ESA_14062021_A_3.c` affinché, ad un preciso istante dell'esecuzione del codice, la memoria centrale si trovi nello stato descritto dall'immagine nel file `ESA_14062021_A_3.pdf`. L'immagine rappresenta una lista di interi concatenata allocata in memoria dinamica. Si inserisca un commento nel punto preciso del codice in cui tale stato si manifesta. La funzione `genera_lista_din()` deve essere utilizzata per la generazione della lista e deve essere implementata in stile **ricorsivo**.