



<b>Cognome</b>	<b>Nome</b>	<b>Matricola</b>	<b>Voto: ... /30</b>
----------------	-------------	------------------	----------------------

INIZIARE LA SOLUZIONE DI OGNI  
 ESERCIZIO SU UNA PAGINA NUOVA

Quesito:	1	2	3	4	5	6	7	Tot.
Max:	4	5	5	5	2	5	4	30
Punti:								

RESTITUIRE COMPILATO ANCHE  
 NEL CASO IN CUI CI SI RITIRA

**Istruzioni:**

- non è possibile consultare libri, appunti, la calcolatrice o qualsiasi dispositivo elettronico, né comunicare;
- si può scrivere con qualsiasi colore, anche a matita, ad eccezione del rosso.
- tempo a disposizione: 2h

**Stile del codice C:**

- non è necessario inserire direttive `#include`;
- è consentito l'utilizzo di funzioni di libreria.

**Quesito 1 (4 punti)**

Punteggio ottenuto: .../4

Dati i due numeri  $A = -A1_{16MS}$  e  $B = 101010001_{2C2}$  effettuare la conversione in base 2, notazione complemento a 2 (2C2), sul numero minimo di bit necessari a rappresentare entrambi gli operandi. Si effettuino quindi le operazioni  $A+B$  e  $A-B$  indicando esplicitamente se si verifica overflow o meno, e motivando la risposta. Mostrare i passaggi fatti.

**Quesito 2 (5 punti)**

Punteggio ottenuto: .../5

Un numero  $n$  è detto "troncabile primo a sinistra" se  $n$  e tutti i numeri ottenuti rimuovendo successivamente la cifra più a sinistra sono primi. Scrivere il sottoprogramma `lefttprime` che ricevuto in ingresso un intero `val` restituisce 1 se il numero è troncabile primo a sinistra.

**Esempio:**

**Ingresso:** 317  
 317 è primo -> 17 è primo -> 7 è primo  
**Uscita:** 1

**Quesito 3 (5 punti)**

Punteggio ottenuto: .../5

Scrivere un sottoprogramma **ricorsivo** che ricevuto in ingresso un array di interi (e qualsiasi altro parametro ritenuto strettamente necessario) restituisce il numero di "vette" in esso presenti. Si definisce "vetta" un valore presente nell'array maggiore di tutti i valori successivi presenti nell'array.

**Quesito 4 (5 punti)**

Punteggio ottenuto: .../5

Scrivere un sottoprogramma che ricevuto in ingresso il nome di un file (comprensivo di percorso ed estensione) contenente date espresse nel formato AAAA MM GG calcola e trasmette al chiamante le date riferite al primo giorno e all'ultimo giorno presenti nel file. Le date nel file non sono ordinate in alcun modo, il file contiene almeno una data, le date inserite sono senz'altro valide e non è necessario fare verifiche. Si utilizzi il tipo di dato di seguito definito.

```
typedef struct _d {
    int year;
    int month;
    int day;
} date_t;
```

**Esempio:**

**contenuto del file:** 2016 02 14 2016 02 11 2015 12 31 2016 03 12 2015 04 03 2015 09 11 2016 03 11  
**ingresso:** ./date.txt  
**uscita:** 2015 04 03 2016 03 12

**Quesito 5 (2 punti)**

Punteggio ottenuto: .../2

Scrivere un programma che acquisito il nome del file da riga di comando, chiama il sottoprogramma sviluppato al punto precedente e visualizza - in base al contenuto del file - il messaggio seguente (il giorno ed il mese non sono necessariamente su due cifre):

```
periodo dal g-m-AAAA al g-m-AAAA
```

**Esempio:**

```
ingresso: ./date.txt
```

```
uscita: periodo dal 3-4-2015 al 12-3-2016
```

**Quesito 6 (5 punti)**

Punteggio ottenuto: .../5

Scrivere un sottoprogramma che riceve in ingresso due liste per la gestione di numeri interi, ciascuna delle quali ordinata in ordine crescente, crei una nuova lista contenente l'unione dei valori presenti nelle due liste, priva di ripetizioni e la restituisca al chiamante.

**Esempio:**

```
contenuto dell'array 10 4 3 5 1 2 3 1 1 1
```

```
uscita: 3 corrispondente ai valori evidenziati
```

**Quesito 7 (4 punti)**

Punteggio ottenuto: .../4

Qual è l'architettura logica del bus di sistema?