



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

INIZIARE LA SOLUZIONE DI OGNI  
 ESERCIZIO SU UNA PAGINA NUOVA

RESTITUIRE COMPILATO ANCHE  
 NEL CASO IN CUI CI SI RITIRA

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

**Istruzioni:**

- la prova è superata se si consegue una valutazione non inferiore a 18/30;
- 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: 1h 45m

**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  
 Scrivere un sottoprogramma `get_id` che ricevuta in ingresso una stringa `link` ed una stringa `header`, trova e restituisce al chiamante l'identificativo numerico intero `id` presente nella stringa `link`, dopo la parte iniziale individuata dall'`header`.

**Esempio:**

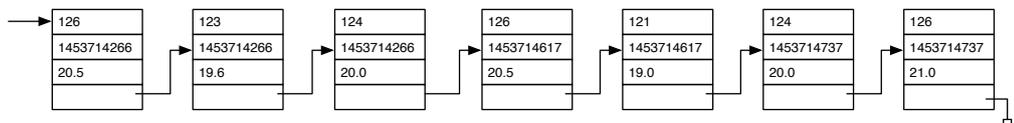
**Ingresso:** `link:http://131.175.56.243/measure/60min/sensor/variable/400/2016/01/29`  
`header:http://131.175.56.243/measure/60min/sensor/variable/`  
**Uscita:** 400

**Quesito 2 (5 punti)** Punteggio ottenuto: .../5  
 Scrivere un sottoprogramma che ricevute in ingresso due stringhe restituisce 1 se sono una l'anagramma dell'altra, 0 altrimenti. Il sottoprogramma non deve modificare le stringhe ricevute in ingresso.

**Quesito 3 (5 punti)** Punteggio ottenuto: .../5  
 Scrivere un sottoprogramma che ricevuta in ingresso una lista per la gestione di informazioni strutturate (il cui tipo è riportato di seguito) **visualizza** per ogni diverso valore del campo `ufficio` presente nella lista, quanti elementi compaiono nella lista.

```
typedef struct _elem {
    int ufficio;
    char * timestamp;
    int valore;
    struct _elem * next;
} data_t;
```

**Esempio:**



**uscita:** 126 3 123 1 121 1 124 2

**Quesito 4 (6 punti)** Punteggio ottenuto: .../6  
 Scrivere un sottoprogramma che ricevuta in ingresso una lista per la gestione di numeri interi, restituisce la lista in modo tale che tutti gli elementi adiacenti uguali siano ridotti ad un solo elemento. Scrivere anche la definizione del tipo opportuno.

**Esempio:**

**contenuto della lista:** 1 → 2 → 3 → 3 → 3 → 2 → 2 → 1 → 4 → 4 → 0 → 0 → 7  
**lista restituita:** 1 → 2 → 3 → 2 → 1 → 4 → 0 → 7

**Quesito 5 (4 punti)**

Punteggio ottenuto: .../4

Scrivere un sottoprogramma **ricorsivo** che ricevuti in ingresso due interi  $n$  e  $k$  con  $n \geq k \geq 0$  senz'altro, calcola e restituisce al chiamante il coefficiente binomiale

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (n \geq k)$$

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$$

**Quesito 6 (6 punti)**

Punteggio ottenuto: .../6

Scrivere un sottoprogramma che ricevuta in ingresso una stringa, restituisce una struttura dati che contenga (con una opportuna gestione della memoria, priva di sprechi) un insieme di stringhe di 160 caratteri ciascuna, contenenti le parti della stringa iniziale suddivisa in pezzi da 160 caratteri. Ad esempio, se la stringa iniziale contiene la frase seguente:

Il testo di questa stringa ha una lunghezza complessiva di 331 caratteri, per cui verra' spezzato in 3 stringhe di 160 caratteri ciascuna per consentire l'invio di 3 sms distinti. E' necessario predisporre una opportuna struttura dati in grado di contenere esattamente le 3 stringhe di al piu' 160 caratteri, senza sprechi.

**il sottoprogramma dovrà creare e restituire 3 stringhe, cosiffatte:**

Il testo di questa stringa ha una lunghezza complessiva di  $x$  caratteri, per cui verra' spezzato in 3 stringhe di 160 caratteri ciascuna per consentire l'invio d

i 3 sms distinti. E' necessario predisporre una opportuna struttura dati in grado di contenere esattamente le 3 stringhe di al piu' 160 caratteri, senza sprechi di memoria.

**Il sottoprogramma restituisce e non visualizza nulla.**