

Fondamenti di Informatica - A.A. 2016-2017

Prof. Vincenzo Caglioti

Appello del **23/02/2017****POLITECNICO**
MILANO 1863

Cognome	Nome	Matricola	Voto: .../30
----------------	-------------	------------------	---------------------

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

Istruzioni:

- per contribuire alla valutazione finale, è necessario conseguire almeno 18/30;
- non è possibile consultare libri, appunti, la calcolatrice o qualsiasi dispositivo elettronico, né comunicare;
- tempo a disposizione: 2h 00m

Stile del codice C:

- non è necessario inserire direttive `#include`;
- i commenti non sono necessari, ma potrebbero essere utili in caso di errore;
- è consentito l'utilizzo di funzioni di libreria.

INIZIARE LA SOLUZIONE DI OGNI
ESERCIZIO SU UNA PAGINARESTITUIRE COMPILATO ANCHE
NEL CASO IN CUI CI SI RITIRA**Quesito 1 (5 punti)**

Punteggio ottenuto .../5

Dati i due numeri $A = +57$ in base 10 e $B = +3A$ in base 16, 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 verifici o meno overflow e motivando la risposta. Mostrare i passaggi svolti.

Quesito 2 (5 punti)

Punteggio ottenuto .../5

Scrivere un programma per il calcolo di statistiche sui caratteri presenti in un file di testo il cui nome è specificato come primo parametro sulla riga di comando. Il file contiene un testo, suddiviso in una o più linee di lunghezza pari al più a 100 caratteri cadauna. Ogni linea è terminata da un carattere 'a capo'.

Nel calcolo delle statistiche, il programma considera i caratteri letti dal file AD ECCEZIONE DEGLI SPAZI E DEGLI 'A CAPO'.

Esso fornisce in uscita:

- il numero di linee presenti nel file;
- il numero totale di caratteri;
- il numero massimo e il numero medio di caratteri per linea;
- il testo completo della linea più lunga presente nel file.

Ad esempio, dato il file contenente il seguente testo:

```
Prova di contenuto di file
per avere un esempio.
```

```
Non e' importante quello che c'e' scritto
presente.
```

il programma visualizza:

```
Linee: 4
```

```
Car: 84
```

```
Max, Med: 35, 21
```

```
Linea max: Non e' importante quello che c'e' scritto
```

Quesito 3 (3 punti)

Punteggio ottenuto .../3

Si consideri il seguente programma C. Allo studente è richiesto di:

1. sostituire il proprio numero di matricola alle cifre tra virgolette nella istruzione indicata con `@@@`;
2. indicare che cosa stampa a schermo il programma.

```
#include <stdio.h>
#define MAX 100
```

```
void M(char *S1, char *S2, char *S3);
```

```
int main()
{
```

```

char A[MAX] = "test";
char B[MAX] = "000000"; /* @@@ */
char C[2*MAX];
M(A, B, C);
printf("%s", C);
return 0;
}

void M(char *S1, char *S2, char *S3)
{
    if (('\'0' == *S1) && ('\'0' == *S2))
    { *S3 = '\0';
      return;
    }

    if ('\'0' != *S1)
    { *S3 = *S1;
      ++S1;
      ++S3;
    }

    if ('\'0' != *S2)
    { *S3 = *S2;
      ++S2;
      ++S3;
    }

    M(S1, S2, S3);
}

```

Quesito 4 (5 punti)

Punteggio ottenuto .../5

Scrivere un sottoprogramma ricorsivo che ricevuto in ingresso un numero intero restituisca la cifra più alta della rappresentazione decimale.

Ad esempio, se il valore ricevuto in ingresso è milleduecentotrentadue il sottoprogramma restituisce 3.

Quesito 5 (5 punti)

Punteggio ottenuto .../5

Una Parola è rappresentata come una stringa di al più 20 caratteri compreso il carattere di terminazione. La ListaOccorrenze, associata a una Parola, è una lista sequenziale -di al più 19 elementi- ognuno dei quali contiene un carattere facente parte della parola e il numero di volte in cui questo carattere compare nella parola.

Si definiscano il tipo Parola e il tipo ListaOccorrenze. Si scriva poi una procedura ricorsiva che, ricevendo in ingresso una Parola, costruisca la ListaOccorrenze associate alla parola ricevuta in ingresso. L'ordine con cui i vari caratteri devono essere posizionati nella ListaOccorrenze deve essere lo stesso con cui essi compaiono per la prima volta nella Parola.

Ad esempio:

la ListaOccorrenze associata alla parola `ordini` è `[5, ['o',1] ['r',1] ['d',1] ['i',2] ['n',1] [...]]`
 mentre la ListaOccorrenze associata alla parola `amaca` è `[3, ['a',3] ['m',1] ['c',1] [...]]`.

Quesito 6 (7 punti)

Punteggio ottenuto/7

Si consideri ancora il problema affrontato nel Quesito 5 con le seguenti differenze.

- La ListaDinamicaOccorrenze -associata a una Parola- sia una lista **dinamica**, ciascun elemento della quale contiene:
 - un carattere facente parte della Parola,
 - il numero di volte in cui questo carattere appare nella Parola,
 - il puntatore al successivo elemento della lista dinamica.
- L'ordine con cui i vari caratteri appaiono nella ListaDinamicaOccorrenze sia l'ordine **alfabetico**.

Ad esempio:

La ListaDinamicaOccorrenze associata alla parola `ordini` è $L \rightarrow ('d',1) \rightarrow ('i',2) \rightarrow ('n',1) \rightarrow ('o',1) \rightarrow ('r',1) \rightarrow \text{NULL}$,
 mentre la ListaDinamicaOccorrenze associata alla parola `amaca` è $L \rightarrow ('a',3) \rightarrow ('c',1) \rightarrow ('m',1) \rightarrow \text{NULL}$.

Si definisca il tipo ListaDinamicaOccorrenze. Si scriva poi una procedura **non necessariamente ricorsiva** che, ricevendo in ingresso una Parola, costruisca la ListaDinamicaOccorrenze associata alla parola ricevuta in ingresso.