	Politecnico di Milano Scuola di Ingegneria Industriale e dell'Informazione FONDAMENTI DI INFORMATICA Appello 11 Settembre 2017		COGNOME E NOME						
	RIGA	COLONNA	MATRICOLA						
<div style="text-align: right;">Spazio riservato ai docenti</div> <table border="1" style="margin-left: auto;"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>									

- Il presente plico contiene 4 esercizi e **deve essere debitamente compilato con cognome e nome, numero di matricola.**
- Il tempo a disposizione è di 1 ore e 20 minuti.
- Non separate questi fogli. Scrivete la soluzione solo sui fogli distribuiti, utilizzando il retro delle pagine in caso di necessità. Cancellate le parti di brutta con un tratto di penna.
- Ogni parte non cancellata a penna sarà considerata parte integrante della soluzione.
- È possibile scrivere a matita (e non occorre ricalcare al momento della consegna!).
- **È vietato utilizzare telefoni o pc.** Chi tenti di farlo vedrà annullata la sua prova.
- È ammessa la consultazione di libri e appunti.
- **Qualsiasi tentativo di comunicare con altri studenti comporta l'espulsione dall'aula.**
- È possibile ritirarsi senza penalità.
- **Non è possibile lasciare l'aula conservando il tema della prova in corso.**

Esercizio 1 (12 punti)

Scrivere una funzione C che riceve come parametro un numero intero n. Il sottoprogramma calcola e restituisce le coppie di interi positivi la cui somma è pari a n.

Note: utilizzare la struttura dati più adatta per restituire le coppie di interi positivi a somma n; l'ordine degli interi restituiti in ciascuna coppia è irrilevante: se viene restituita la coppia (x,y) non occorre restituire anche la coppia (y,x).

Esempi

ingresso: 8; uscita: (1,7) (2,6) (3,5) (4,4)

ingresso: 11; uscita: (1,10) (2,9) (3,8) (4,7) (5,6)

Soluzione

```
typedef struct
{
    int x;
    int y;
} coppia;

coppia* somme(int n)
{
    coppia *c = malloc(sizeof(coppia)*n/2);
    for (int i=0; i<n/2; i++)
    {
        c[i].x = i+1;
        c[i].y = n-i-1;
    }
    return c;
}
```

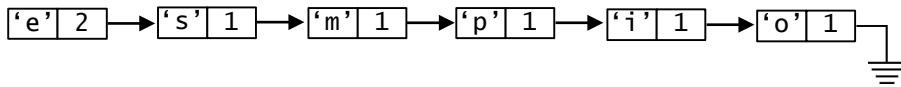
Esercizio 2 (12 punti)

Si implementi una funzione `string2list` che riceve in ingresso una stringa `str` e restituisce una lista così organizzata: la lista dovrà contenere un elemento per ciascun carattere **distinto** presente in `str`; inoltre, ogni elemento della lista dovrà contenere sia il carattere associato a tale elemento, che il numero di volte in cui tale carattere compare all'interno della stringa `str`.

Esempio

Ingresso: "esempio"

Uscita:



Soluzione

```
struct nodo
{
    char c;
    int n;
    struct nodo *next;
};

typedef struct nodo *lista;

lista string2list(char str[])
{
    lista l = NULL;
    int n = strlen(str);

    for (int i=0; i<n; i++)
    {
        inserisci(&l, str[i]);
    }

    return l;
}

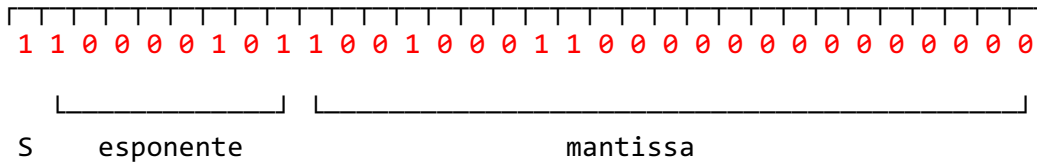
void inserisci(lista *l, char c)
{
    if (*l==NULL)
    {
        *l = malloc(sizeof(struct nodo));
        (*l)->c = c;
        (*l)->n = 1;
        (*l)->next = NULL;
    }
    else if ((*l)->c == c)
        (*l)->n++;
    else
        inserisci(&((*l)->next), c);
}
```

Esercizio 3 (4 punti)

Si consideri il numero negativo e razionale -100.375

Riportare nello spazio seguente la sua codifica secondo lo standard IEEE a precisione singola (riportare di seguito anche i calcoli effettuati).

Soluzione



S: 1

$$100 = 4 + 32 + 64 \rightarrow 1100100$$

$$0.375 * 2 = 0.75$$

$$0.75 * 2 = 1.5$$

$$0.5 * 2 = 1.0$$

$$0.375 \rightarrow 0.011$$

$$1100100.011 = 1.100100011 * 2^6$$

$$E = 6 + 127 = 133 = 128 + 4 + 1 = 10000101$$

S: 1

M: 1001000110 0000000000 000

E: 10000101

Esercizio 4 (4 punti)

Un sistema ha una memoria con le seguenti caratteristiche: memoria fisica di 32 Mbyte e memoria virtuale di 64 Mbyte con pagine di dimensione 32Kbyte.

Rispondere alle seguenti domande giustificando le risposte:

- A. Quale è la struttura dell'indirizzo virtuale e di quello fisico, e la lunghezza dei campi che li costituiscono?
- B. Un consulente afferma che raddoppiando il numero di pagine virtuali di memoria è possibile ridurre gli sprechi di memoria. Sei d'accordo con questa affermazione? Giustificare la risposta.

Soluzione

- A. 32 Mbyte di memoria fisica indirizzabile \rightarrow 25 bit di indirizzo fisico
64 Mbyte di memoria virtuale indirizzabile \rightarrow 26 bit di indirizzo virtuale
Pagina di 32Kbyte = 2^{15} byte \rightarrow 15 bit offset

Quindi si ha,

Indirizzo virtuale: NPV: 11 bit, offset: 15 bit

Indirizzo fisico: NPF: 10 bit, offset: 15 bit

- B. Sono d'accordo, dal momento che raddoppiando il numero di pagine virtuali, si dimezzerà la dimensione della pagina virtuale e quindi si ridurranno possibili sprechi di memoria.