UNIVERSITÀ DEGLI STUDI DI PISA

FACOLTÀ DI INGEGNERIA

CORSO di LAUREA in INGEGNERIA delle TELECOMUNICAZIONI

ESAME DI FONDAMENTI DI INFORMATICA 12/01/2009 - C++

1. **Sia dato un file** `coppie1.dat` **contenente in ciascuna riga una coppia di valori interi. Si scriva una funzione** `genera_file()` **che calcola il valore massimo** `M_c` **e minimo** `m_c` **delle somme delle coppie di numeri del file** `coppie1.dat`**. Inoltre la funzione crea un nuovo file** `coppie2.dat` **che contiene le sole coppie di** `coppie1.dat` **la cui somma è minore di** `M_c – m_c`**.**

2. **Si scriva una funzione** `inverti(s)` **che riceve in ingresso una stringa di lunghezza massima 24 caratteri e restituisce la stringa invertita**.
   `[es. se s="compito" => s invertita = "otipmoc"]`

3. **Sia data una lista circolare doppia a valori interi** `l`**. Si scriva una funzione** `elimina (l, n)` **che riceve in ingresso la lista** `l` **e un valore intero** `n` **e ne elimina l'elemento che si incontra scorrendola di** `n` **passi.**

4. **Dati i numeri decimali** `A=124`**,** `B=101` **e** `C=-87`**, si determini il minimo numero di bit necessari per rappresentare contemporaneamente i suddetti numeri in complemento a 2 e se ne dia la rappresentazione. Utilizzando poi lo stesso numero di bit, si esegua l'operazione** `D=B-C` **e si discuta se il risultato ottenuto è o no significativo.**

Punteggio:

Es.1	Es.2	Es.3	Es.4
10	10	10	3

Risultati e Soluzione del compito:http://www.isti.cnr.it/~glami/elenco-compiti.htm

1.

```
#include <fstream.h>

void genera_file() {
int n1, n2;
int M_c, m_c;
fstream f, g;
f.open("coppie1.dat", ios::in);
g.open("coppie2.dat", ios::out);
if(f>>n1>>n2) {
        M_c=n1+n2;
        m_c=n1+n2;
        }
else return;
while(f>>n1>>n2) {
        if(n1+n2 >M_c) M_c=n1+n2;
        if(n1+n2<m_c) m_c=n1+n2;
        }
f.close();
f.open("coppie1.dat", ios::in);
while(f>>n1>>n2) {
        if(n1+n2 < (M_c – m_c)) g<<n1<<" "<<n2<<"\n";
        }
f.close();
g.close(); }
```

2.

```
char* inverti (char s[25]) {
char x;
int cont=0;
for(int i=0; s[i]!='\0'; i++)
        cont++;
for(int i=0; i<cont/2; i++) {
        x=s[i];
        s[i]=s[cont-i-1];
        s[cont-i-1]=x;
        }
reutrn s; }
```

3.

```
struct elem
{       int inf;
        elem* next;
        elem *prec; };

void elimina (int n, elem *&l)
{
elem *aux=l;
for(int i=0 ; i<n ; i++)
        aux=aux->next;
if(aux==l)
        l=aux->next ;
aux->prec->next=aux->next;
aux->next->prec=aux->prec ;
delete aux;
}
```