



UNIVERSITÀ DEGLI STUDI DI PISA
FACOLTÀ DI INGEGNERIA
CORSO di LAUREA in INGEGNERIA delle TELECOMUNICAZIONI

ESAME DI FONDAMENTI DI INFORMATICA - SIMONCINI – 13/7/2006 - C++

1. Un ufficio postale ha un dispositivo che segnala se uno dei suoi sportelli è libero, occupato oppure chiuso. Tale dispositivo può essere rappresentato mediante un array di caratteri dove l'elemento i -esimo vale C se l' i -esimo sportello è chiuso, L se è aperto e libero, B se è aperto e occupato. Si definisca una classe `posta` assumendo che le possibili operazioni siano le seguenti:
 - `posta(n)`: costruttore che inizializza un dispositivo con n sportelli chiusi ($n=10$ per default);
 - `ok=Liberato(i)`: se lo sportello di indice i è chiuso o occupato lo rende libero e restituisce T , altrimenti restituisce F ;
 - `ok=Occupato(i)`: se lo sportello di indice i è libero lo occupa e restituisce T , altrimenti restituisce F ;
 - `n = conteggio()`: restituisce in uscita il numero degli sportelli chiusi oppure occupati;
 - `~posta()`: distruttore;
2. Un file chiamato `numeri.dat` contiene un insieme non vuoto di numeri interi. Scrivere una funzione `cerca_estremi()` che legge il file e stampa sul monitor il numero più grande e quello più piccolo contenuti in esso.
3. Si scriva una funzione `rendi_pari(l)` che riceve in ingresso il puntatore ad una lista semplice a valori di tipo carattere l . La funzione conta il numero n di elementi di cui è composta l e, se n è dispari, allora viene eliminato l'elemento centrale di l , altrimenti l rimane invariata.
4. Dati i numeri decimali $A=-65$, $B=63$ e $C=126$, 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 $D=A+C+B$ e si discuta se il risultato ottenuto è o no significativo.

Punteggio:

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



UNIVERSITÀ DEGLI STUDI DI PISA
FACOLTÀ DI INGEGNERIA
CORSO di LAUREA in INGEGNERIA ELETTRONICA

Es. 1

```
// posta.h
enum ind {C, B, L};
class posta {
    ind *D;
    int N=10;
public:
    posta() ;
    posta(int n);
    bool liberato(int i) ;
    bool occupato (int i);
    int conteggio ()
    ~posta() {delete [ ] D;};
}

// posta.cc
#include "posta.h"

posta::posta(int n)
{
    N=n;
    D = new int[N];
    for (int i=0; i<N; i++)
        D[i]='C';
}

posta::posta()
{ D = new int[N];
  for (int i=0; i<N; i++)
    D[i]='C';
}

bool posta::liberato(int i) {
    if (D[i]=='B')||(D[i]=='C')
        {
            D[i]='L';
            return T; }
    return F;
}

bool posta::occupato(int i)
{
    if (D[i]=='L')
    { D[i] = 'B';
      return T; }
    return F;
}

int posta::conteggio()
{
    int cont=0;
    for (int i=0; i<N;i++)
    if (D[i]!='C')
        cont++;
    return cont;
}
```



UNIVERSITÀ DEGLI STUDI DI PISA
FACOLTÀ DI INGEGNERIA
CORSO di LAUREA in INGEGNERIA ELETTRONICA

Es. 2

```
#include <fstream.h>
void cerca_estremi() {
    fstream f;
    f.open("numeri.txt", ios::in);
    int letto;
    int max, min, cross;
    f>>min;
    if(f>>max);
    else { cout<< "un solo valore: "<<min;
           return; }
    if(max<min ) { cross=max;
                  max=min;
                  min=cross;
    }
    while(f>>letto) {
        if(letto>max) max=letto;
        else if(letto<min) min=letto;
    }
    cout<<"massimo: "<<max<<"\nminimo: "<<min;
    f.close(); }
```

Es. 3

```
void rendi_pari(elem *&l) {
    int cont=0;
    if (l==0) return;
    if(l->next==0) { delete l;
                    l=0;
                    return; }
    elem *aux=l;
    while(aux!=0) { cont++;
                    aux=aux->next; }
    if(cont%2==0) return;
    aux=l;
    elem *aux1=aux->next;
    for(int i=cont/2-1; i!=0; i--)
    { aux=aux1;
      aux1=aux1->next; }
    aux->next=aux1->next;
    delete aux1; }
```