



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

ESAME DI FONDAMENTI DI INFORMATICA - SIMONCINI – 17/07/2008 - C++

1. Si definisca una classe `coda` che permetta di eseguire le seguenti operazioni su una lista semplice a valori interi `l`:
 - `coda()`: costruttore che crea una lista vuota;
 - `inserimento (info)`: inserisce un elemento con campo informativo uguale a `info` in fondo alla lista;
 - `boolean estrazione()`: se la lista non è vuota, elimina dalla lista il primo elemento restituendo `T`, altrimenti restituisce `F`;
 - `~coda()`: distruttore che dealloca la lista.

2. Sia dato il file `insieme_coppie.dat` che contiene un insieme di coppie di valori interi. Si scriva una funzione `crea_file(x)` che riceve in ingresso un intero `x` e crea un nuovo file `F1.dat` che contiene quelle coppie di interi contenute in `insieme_coppie.dat` la cui somma è maggiore di `x`.

3. Dati un vettore `A1` a valori interi di dimensione `N`, si scriva una funzione `stampa_Mm(A1)` che riceve in ingresso `A1` e stampa sul monitor l'elemento più grande e quello più piccolo in esso contenuti.

4. Dati i numeri decimali $A=113$, $B=42$ e $C=-89$, 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
11	9	10	3



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

1.

```
//coda.h
```

```
class coda {  
    elem* l;  
public:  
    coda() {l=0};  
    void inserimento (int info);  
    boolean estrazione ();  
    ~coda();}
```

```
//coda.cc  
#include "coda.h"
```

```
void coda::inserimento (int n)
```

```
{    elem* aux;  
    if (l==0) {  
        l=new elem;  
        l->info=n;  
        l->next=0;  
        return;}  
    for (aux=l; aux->next !=0 ; aux=aux->next) ;  
  
    elem np;  
    new elem np ;  
    np->info = n;  
    np ->next=0;  
    aux->next=np ;  
}
```

```
bool coda::estrazione ()
```

```
{    elem* aux=l;  
    if (l==0) return F;  
    l=l->next;  
    delete aux;  
    return T;}
```

```
coda::~~coda() {  
    elem* aux=l;  
    while (aux!=0) {  
        l=l->next;  
        delete aux;  
        aux = l;}  
}
```

2.

```
#include <fstream.h>  
void crea_file( int x) {  
    fstream f, g;  
    f.open("insieme_coppie.dat", ios::in);  
    g.open("F1.dat", ios::out);  
    while(f>>n>>m)  
        if(n+m >x) g<<n<<" "<<m<<"\n";  
    f.close();
```



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

```
g.close();  
}
```

3.

```
#include <iostream.h>
```

```
const int N=10 ;
```

```
void stampa_Mm (int A1[N])
```

```
{
```

```
    int min, max;
```

```
    max=A1[0];
```

```
    min=A1[0];
```

```
    for(int i=1; i<N; i++) {
```

```
        if(A1[i]>max) max=A1[i];
```

```
        if(A1[i]<min) min=A1[i];
```

```
    }
```

```
    cout<<max;
```

```
    cout<<min;
```

```
}
```