

```

/*
Classe : Bussolotto

Autore : Riontino Raffaele 4 informatici (corso serale)
        ITIS Molinari - Milano
        2/03/2011

Funzionalità : bussolotto per la generazione e la gestione di numeri
               casuali da utilizzare con il gioco della tombola , superenalotto,
               o altri giochi simili
*/

#include <iostream>
#include <vector>
#include <time.h>
using namespace std;

//srand((unsigned)time(NULL));

typedef vector<bool> boolVet;

typedef vector<int> intVet;

class Bussolotto
{
protected : boolVet numeri; //contiene i numeri totali del bussolotto
              intVet sequenza; //salva la sequenza dei numeri estratti
              int estratti ; //contiene quanti numeri sono stati estratti

public :
    Bussolotto (int); //costruttore che ha come parametro il numero massimo es.90
    int estrai (); //estrae un numero casuale tra quelli presenti nel vector 'numeri'
    void resetta (); //rigenera il bussolotto
    void resetta (int); //rigenera il bussolotto con la possibilità di cambiare il
limite massimo
    void visualizzaEstratti (); //visualizza tutti i numeri estratti
    void visualizzaTabellone (); //visualizza il tabellone come quello della tombola
    int numeriEstratti (); //ritorna quanti numeri sono stati estratti
    bool controllaNumero (int); //controlla se un numero è già stato estratto
};

/////definizione metodi c

Bussolotto::Bussolotto(int dim)
{
    numeri.resize(dim,false);
    this -> estratti = 0;
    //srand((unsigned)time(NULL));
}

int Bussolotto::estrai()
{
    if (estratti == numeri.size())
        return -1; //se sono già stati estratti tutti i numeri ritorna -1
    int temp = rand()%numeri.size();
    while(numeri.at(temp))
    {
        if (temp == (numeri.size()-1)) temp = 0;
        else temp++;
    }
    numeri.at(temp) = true;
    estratti ++;
    sequenza.push_back(temp+1);
}

```

```

    return (temp + 1);
}

void Bussolotto::resetta ()
{
    int temp = numeri.size();
    numeri.clear(); //elimino tutte le posizioni del vettore
    sequenza.clear();
    numeri.resize(temp , false); // ridefinisco la dimensione del vettore e lo inizializzo a
false
    this -> estratti = 0;
}

void Bussolotto :: resettta (int dim)
{
    numeri.clear();
    sequenza.clear();
    numeri.resize(dim , false);
    this -> estratti = 0;
}

void Bussolotto :: visualizzaTabellone ()
{
    int conto = 0 ;
    cout << "\n\t\tTabellone da " << numeri.size() << " numeri\n\n";
    for (int i = 0 ; i < numeri.size() ; i++)
    {
        if (numeri.at(i))
        {
            if ((i+1) < 10) cout << " ";
            cout << " " << (i+1);
        }
        else cout << " X";
        conto++;
        if (conto == 10)
        {
            conto = 0;
            cout << endl;
        }
        if (conto == 5) cout << " ";
    }
}

void Bussolotto :: visualizzaEstratti ()
{
    int conto = 0 ;
    cout << "\n\t\tNumeri estratti\n\n";
    for (int i = 0 ; i < sequenza.size() ; i++)
    {
        if ((i+1) < 10) cout << " ";
        cout << " " << sequenza.at(i);

        conto++;
        if (conto == 10)
        {
            conto = 0;
            cout << endl << endl;
        }
    }
}

int Bussolotto :: numeriEstratti ()
{
    return estratti;
}

bool Bussolotto :: controllaNumero (int num)
{
    if (numeri.at(num-1)) return true;
}

```

```
else return false;  
}
```