

```

/*
Programma : cppVector

Autore : Riontino Raffaele 4 informatici (serale)
        ITIS Molinari - Milano 18/01/2011

Funzionalità : esempio di utilizzo del contenitore vector
                appartenente alla "Standard Template Library",
                e di alcuni dei suoi metodi.

*/

#include <iostream>
#include <conio.h>
#include <vector>
#define SIZE 0 //dimensione del contenitore
using namespace std;

vector <int> v(SIZE); //dichiarazione di un contenitore di tipo vector con dimensione SIZE
vector<int>::iterator it; //dichiarazione di un iteratore (puntatore) al contenitore

char menu();

void inserisciCoda();

void eliminaCoda();

void inserisciMezzo();

void cambiaDimensione();

void visualizza();

void eliminaMezzo();

int main()
{
    char scelta;
    do{
        system("cls");
        cout << "\n\tProve sull'utilizzo dell'oggetto vector\n\n";
        cout << "\n\tCapacità : " << v.capacity(); //visualizzo la capacità del vettore
        cout << "\tDimensione : " << v.size(); //visualizzo la dimensione del vettore
        scelta = menu();
        switch (scelta)
        {
            case 'i' : inserisciCoda();
                       break;
            case 'd' : eliminaCoda();
                       break;
            case 'm' : inserisciMezzo();
                       break;
            case 'v' : visualizza();
                       break;
            case 'c' : eliminaMezzo();
                       break;
            case 'r' : cambiaDimensione();
                       break;
            case 'e' : break;
        }
    }while(scelta != 'e');
    return 1;
} //fine main

```

```

char menu()
{
    char scelta;
    cout << "\n\n\t\tMenu\n\n";
    cout << "\n\t[ i ] - inserisci un elemento in coda v.push_back()\n";
    cout << "\n\t[ d ] - elimina elemento in coda v.pop_back()\n";
    cout << "\n\t[ c ] - elimina elemento in mezzo v.erase(v.begin()+pos)\n";
    cout << "\n\t[ m ] - inserisci un elemento in mezzo (v.insert(v.begin()+pos,dato))\n";
    cout << "\n\t[ v ] - visualizza tutto il contenuto\n";
    cout << "\n\t[ r ] - cambia dimensione v.resize()\n";
    cout << "\n\t[ e ] - esci";
    cout << "\n\n\t digita la scelta : ";
    scelta = getch();
    scelta = tolower(scelta);
    return scelta;
} //chiusura menu

void inserisciCoda()
{
    int* temp = new int;
    cout << "\n\n\t inserisci un intero : ";
    cin >> *temp;
    v.push_back(*temp); //inserimento in coda al vettore
    delete temp;
} //chiusura inserisciCoda

void eliminaCoda()
{
    //controllare prima se il contenitore è pieno
    if (v.empty()) cout << "\n\n\t Contenitore vuoto!!";
    else {
        v.pop_back(); //eliminazione dell'ultimo elemento del vettore
        cout << "\n\n\t Elemento in coda cancellato!!";
    }
    getch();
} //chiusura eliminacoda

void inserisciMezzo()
{
    int* temp = new int;
    cout << "\n\n\t inserisci l'elemento : ";
    cin >> *temp; //dato da inserire nel vector
    int pos=-1; //conterrà la posizione in cui inserire il dato
    for(int i = 0; pos < 0 && i < v.size(); i++) //scorro il vector
        if(*temp < v.at(i)) //se il dato che ho inserito è maggiore di quello trovato
            alla posizione i
                pos = i; //salvo la posizione
    if(pos >= 0) //se è stata trovata la posizione
        v.insert(v.begin()+pos,*temp); //inserisco il dato con il metodo insert
    else
        v.push_back(*temp); //altrimenti in coda
    delete temp;
} //chiusura inserisciMezzo

void cambiaDimensione()
{
    int* temp = new int;
    cout << "\n\n\t Attenzione : se si inserisce una dimensione minore di quella attuale\n";
    cout << "\t i dati in piu' verranno cancellati!!\n";
    cout << "\n\t dimensione attuale : " << v.size();
    cout << "\n\t inserisci la nuova dimensione : ";
    do {
        cin >> *temp;
    } while (*temp < 0); //non puo' avere una dimensione minore di 0
    v.resize(*temp); //ridimensiono il vettore
    delete temp;
} //chiusura cambiaDimensione

```

```

void eliminaMezzo()
{
    if (v.empty())
    {
        cout << "\n\tContenitore vuoto!!";
        getch();
        return;
    }
    int *temp = new int;
    cout << "\n\tinserisci l'elemento da eliminare : ";
    cin >> *temp;
    int pos = -1;
    for (int i = 0 ; pos < 0 && i < v.size() ; i++)
        if (*temp == v.at(i)) pos = i;
    if (pos >= 0) {
        v.erase(v.begin()+pos);
        cout << "\nelemento eliminato";
    }
    else cout << "\nelemento non trovato";
    getch();
} //eliminaMezzo

```

```

void visualizza()
{
    it = v.begin() ;
    cout << endl;
    if (v.empty()) cout << "\n\n\tvector vuoto!!";
    else
    for (int i = 0 ; it != v.end() ; it++ , i++)
        cout << "[" << i << "] = " << *it << "\t";
    getch();
} //chiusura visualizza

```