

```

/*Utilizzando una struttura struct alunno
    { string codice,nome,indirizzo;
    };
caricare un tabella di n alunni trasformando le minuscole in maiuscole e stampare
l'elenco degli alunni in ordine alfabetico con iniziali A-L e l'elenco degli alunni
in ordine alfabetico con iniziali M-Z.
Specifiche: n compreso tra 4 e 20, codice di lunghezza compresa tra 4 e 6*/
#include<iostream>
#include<string>
using namespace std;
struct alunno
    {string codice,nome,indirizzo;
    };
alunno t[21];int n;
void maius(string &s);
void ordinatabella();
int main()
{
    do // controllo che n sia compreso tra 5 e 20
    {system("cls");
    cout<<"Numero alunni [4÷20] : "; // ÷ =ALT 0247
    cin>>n;
    } while(n<4 || n>20);

    for(int i=1;i<=n;i++)
    { do //controllo che la lunghezza del codice sia compresa tra 4 e 6
    { cout<<"\nCodice alunno "<<i<<" [lunghezza 4÷6] : ";
    cin>>t[i].codice;
    } while(t[i].codice.size()<4 || t[i].codice.size()>6);
    maius(t[i].codice); // converto il codice in maiuscolo
    cout<<"Nome e cognome : ";
    fflush(stdin); getline(cin,t[i].nome);
    maius(t[i].nome); // converto il nome in maiuscolo
    cout<<"Indirizzo : ";
    fflush(stdin); getline(cin,t[i].indirizzo);
    maius(t[i].indirizzo); // converto l'indirizzo in maiuscolo
    }
    ordinatabella();
    cout<<"\n\nElenco degli alunni con iniziali A÷L";
    for(int i=1;i<=n;i++)
    {if(t[i].nome<"M")
    {cout<<"\n\t"<<t[i].codice<<"\t\t"<<t[i].nome<<"\t\t\t"<<t[i].indirizzo;}
    }
    cout<<"\n\nElenco degli alunni con iniziali M÷Z";
    for(int i=1;i<=n;i++)
    {if(t[i].nome>="M")
    {cout<<"\n\t"<<t[i].codice<<"\t\t"<<t[i].nome<<"\t\t\t"<<t[i].indirizzo;}
    }
    cout<<"\n\nF I N E ";
    system("pause");
}
void maius(string &s)
{ for(int i=0;i<=s.size()-1;i++)//converte minuscole in maiuscole
    {if (s[i]>'Z') s[i]=char(s[i]-32);}
}
void ordinatabella()
{ bool ord;
do

```

```
{ord=true;
  for(int i=1;i<=n-1;i++)
    { if (t[i+1].nome<t[i].nome)
      {swap(t[i+1].nome,t[i].nome);
      swap(t[i+1].codice,t[i].codice);swap(t[i+1].indirizzo,t[i].indirizzo);
      ord=false;
      }
    }
  }while(ord==false);
}
```