

Esercitazioni del corso di Laboratorio di Programmazione e Calcolo

n. 2

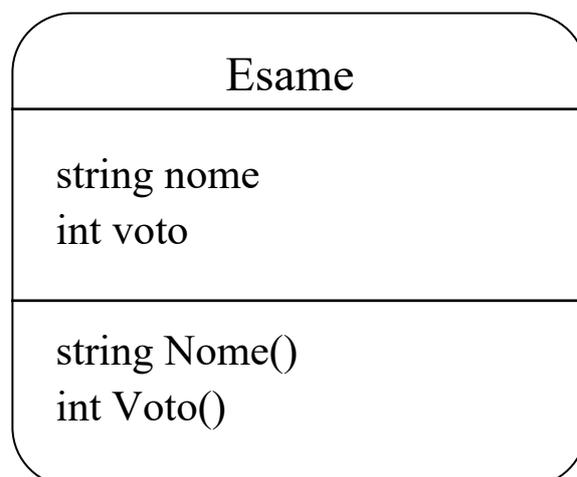
Introduzione alla programmazione in C++: implementazione di una classe Uso di emacs e del compilatore g++

NB: le tastiere italiane (che prevedono caratteri speciali come le lettere accentate) non prevedono caratteri quali { } e ~ che vi serviranno spesso. Con la tastiera italiana utilizzare:

AltGr insieme al tasto 7 per {
AltGr insieme al tasto 0 per }
AltGr insieme al tasto ^ per ~

Lo scopo dell'esercitazione e' di aiutarvi a familiarizzare con la sintassi del linguaggio C++, scrivendo, compilando ed eseguendo un semplice programma. In preparazione alle prossime lezioni, anche se l'argomento non e' stato ancora trattato, proverete a costruire e ad utilizzare una semplice Classe.

1. controllate che nella vostra HOME directory sia presente il file con i vostri nomi. Nel caso non fosse presente, createlo.
2. nella vostra HOME directory create una directory **ex2** che conterra' l'esercitazione numero 2 e spostatevi in ex2
3. costruite un "oggetto" della Classe Esame. L'oggetto e' caratterizzato dal nome dell'esame e dal voto conseguito. L'unica operazione che potete fare su quest'oggetto e' ottenere il nome dell'esame o il voto conseguito



4. per implementare la Classe e' necessario scrivere due files

Esame.h (declaration file)
Esame.cc (implementation file)

Il contenuto del file e' riportato alla fine di questa pagina e verra' discusso in dettaglio a lezione. Oggi, con l'aiuto del docente presente in laboratorio, provate ad "usarlo" e a capire come funziona

5. scrivere un programma main() - che potreste chiamare ad esempio **Esercit_2.cpp** - che esegua le seguenti operazioni

- o istanzi un oggetto di tipo Esame
- o ne stampi su video le caratteristiche (il nome e il voto).
Tecnicamente queste "caratteristiche" si chiamano attributi.

6. Suggerimento 1 : - Per istanziare un oggetto, dovete usare il costruttore. Ad esempio:

```
Esame Test ("test", 30) ;
```

7. Suggerimento 2: - Per eseguire una operazione sull'oggetto (tecnicamente: Per applicare un metodo ad un oggetto), si scrive il nome dell'oggetto, un punto (".") e poi il nome del metodo. Ad esempio:

```
Test.Nome () ;
```

```
Test.Voto () ;
```

8. Compilate la Classe ed il programma Main. Unite (*linkate*) le due parti per ottenere il programma eseguibile, come mostrato al termine della lezione precedente e seguendo l'indicazione del docente esercitatore presente in laboratorio. Per eseguire il programma e' sufficiente scrivere il nome dell'eseguibile, eventualmente preceduto da ./ per indicare che l'eseguibile si trova nella directory corrente (.)

9. (facoltativo) Considerate il programma mostrato a lezione e che calcola il voto finale dopo aver superato sei prove, cioe' sei esami parziali. Trasformate questo programma in un programma ad oggetti, seguendo le seguenti indicazioni:

- Rappresentate le sei prove istanziando sei oggetti della classe Esame.

..- Calcolate il voto finale con l'algoritmo concordato a lezione

- Costruite un oggetto della Classe Esame (che potreste ad esempio chiamare EsameFinale) che corrisponde al corso completo: i valori degli attributi di questo oggetto saranno "Corso di Programmazione e Calcolo" e il voto finale calcolato sopra

- Stampate su video gli attributi dell'oggetto EsameFinale

Esame.h

```
// Nome Cognome

#ifndef ESAME_H
#define ESAME_H

#include <string>
using namespace std;

class Esame {

protected:
    string nome;
    int voto;

public:
    //costruttore e distruttore
    Esame(string nomenow, int votonow);
    ~Esame();

    //metodi di tipo get
    string Nome() ;
    int Voto() ;

};

#endif
```

Esame.cc

```
// Nome Cognome

#include "Esame.h"

#include <string>
using namespace std;

//costruttore e distruttore
Esame::Esame(string nomenow, int votonow){
    nome = nomenow ;
    voto = votonow ; }

Esame::~Esame(){ };

//metodi di tipo get
string Esame::Nome() { return nome; };
int Esame::Voto() { return voto;};
```