

declaration file IdealGas.h

```
#ifndef IDEALGAS_H
#define IDEALGAS_H

#include "Gas.h"

using namespace std ;

class IdealGas : public Gas {

public:

// costruttori

IdealGas() ;
IdealGas(string Name, int nmoli) ;

// distruttore

virtual ~IdealGas() ;

// altri metodi

virtual void print() ;

virtual double
P( double vol, double temp) ;

virtual double
T( double vol, double press) ;

} ;

#endif
```

declaration file RealGas.h

```
#ifndef REALGAS_H
#define REALGAS_H

#include "Gas.h"

using namespace std ;

class RealGas : public Gas {

protected:

double agas ;
double bgas ;

public:

// costruttori

RealGas() ;
RealGas (string Name, int nmoli) ;
RealGas (string Name, int nmoli,
double a, double b) ;

// distruttore

virtual ~RealGas() ;

// altri metodi

virtual void print() ;

virtual double
P( double vol, double temp) ;

virtual double
T( double vol, double press) ;

} ;

#endif
```



Gas Perfetti e Gas Reali

gennaio 2014

- 1) Costruire una classe (che potrete chiamare ad esempio *Gas*) per descrivere il comportamento di un numero *nmoli* di un gas di nome *name*. Tale classe in seguito verrà anche utilizzata, con alcune modifiche, come classe base per altre classi che erediteranno da essa.

In particolare, dovranno essere definiti ed implementati i seguenti metodi

- i) Costruttore di default (*Gas()*), Costruttore con parametri (*Gas (string Name, int n)*) e distruttore (*~Gas()*). Gli argomenti del costruttore sono il nome del gas ed il numero di moli.
- ii) Metodi di tipo Get che forniscano il nome del gas (*Name()*) ed il numero di moli (*Moli()*)
- iii) Un metodo di tipo print (*print()*) che permetta di visualizzare sullo schermo il nome del gas ed il numero di moli.

Si costruisca un semplice programma main per controllare la funzionalità dei metodi implementati. **(5 punti)**

- 2) Si costruiscano due nuove classi *IdealGas* e *RealGas* per lo studio dei gas ideali (o perfetti) e dei gas reali. Le due classi *IdealGas* e *RealGas* ereditano dalla classe *Gas* precedentemente definita.

- i) La classe *RealGas* sia caratterizzata da due ulteriori attributi *agas* e *bgas* che tengano conto dell'equazione dei gas reali (si veda l'equazione di stato riportata in Appendice)
- ii) Si implementino, per ciascuna classe, tutti i Costruttori del tipo definito al punto 1) e, per la classe *RealGas*, il Costruttore con tutti i parametri (*RealGas(string Name, int n, double a, double b)*). Per entrambe le classi si implementino i distruttori.
- iii) Si implementi, nelle due classi, il metodo *print()*, in modo che visualizzi sullo schermo anche il valore dei coefficienti *agas* e *bgas* oppure la scritta "Gas perfetto"
- iv) Si renda virtuale il metodo *print()*

Si modifichi il programma main di test per verificare la correttezza dei metodi implementati **(3 punti)**

3)

Si implementino due metodi virtuali (`double P(double vol, double temp)` e `double T(double press, double press)`) che forniscano, per ciascuna delle due classi, rispettivamente il valore della pressione P e della temperatura T, quando il gas si trova nella configurazione descritta dai valori che sono argomento di ciascun metodo.

Si modifichi il programma main di test per verificare la correttezza dei metodi implementati, per un gas ideale, per l'Azoto e per l'Anidride Carbonica (i corrispondenti valori di a e b sono riportati in Appendice). **(2 punti)**

=====

Suggerimento

I valori di P e di V si ottengono dall'equazione di stato

i) Per un gas perfetto

$$P = \frac{nRT}{V} \qquad T = \frac{PV}{nR}$$

i) Per un gas reale

$$P = \frac{nRT}{(V-nb)} - \frac{n^2a}{V^2} \qquad T = \frac{\left(P + \frac{n^2a}{V^2}\right)(V-nb)}{nR}$$

Ad esempio, l'implementazione del metodo T nella classe RealGas potrebbe essere scritto nella forma:

```
double RealGas::T( double vol, double press) {
    double nmoli2 = nmoli*nmoli ;
    double vol2   = vol*vol      ;
    double temp = press + (nmoli2*agas/vol2) ;
    temp = temp*(vol-nmoli*bgas) ;
    temp = temp/(nmoli*RGAS)      ;
    return temp ;
}
```

APPENDICE

Costante universale dei gas

R = 0.082 litri atm/mole °K = 8,31 J/mole °K

Valori delle costanti a e b per alcuni gas reali

N (azoto)	a = ~ 1,40 litri ² atm /mole ²
	b = ~ 0,030 litri /mole
CO ₂	a = 3.59 litri ² atm /mole ²
	b = 0.0427 litri /mole

Equazione di Stato dei Gas Perfetti (Ideali)

$$PV = nRT$$

Equazione di Stato dei Gas Reali

$$\left(P + \frac{n^2a}{V^2}\right)(V-nb) = nRT$$

declaration file Gas.h

```
#ifndef GAS_H
#define GAS_H

#define RGAS 0.082

#include <iostream>
#include <string>
#include <cmath>

using namespace std ;

class Gas {
protected:
    string name ;
    int nmoli ;

public:
    // costruttori
    Gas() ;
    Gas(string Name, int n) ;

    // distruttore
    virtual ~Gas() ;

    // metodi di tipo Get
    string Name() ;
    int Moli() ;

    // altri metodi
    virtual void print() ;

    virtual double P( double vol, double temp) ;
    virtual double T( double vol, double press) ;

};
```

N.B. Tempo per l'esercitazione: 3 ore (14.30-17.30)

Si possono usare appunti delle lezioni e manuali di C++

La prova d'esonero si intende superata se il programma funziona (compilazione, linking, esecuzione)