

Exercícios de Orientação a Objetos

Observações:

- O trabalho pode ser feito individual, em duplas ou no grupo de trabalho da turma (insira o(s) nome(s) do(s) desenvolvedor(es) como comentário);
- Apenas um envia os arquivos pelo Moodle;
- Use o material de aula, as bibliografias básicas da disciplina para sanar dúvidas;
- Deverá ser entregue no dia 07 de dezembro de 2016, até as 19 horas (G2);
- Organize seu tempo e bom Trabalho!!!

I – Implementar as classes abaixo que sejam capazes de realizar as operações (métodos) apresentadas:

- Classe Círculo:

```
float AreaCirculo(float r);  
float Perimetro(float r);  
int MostrarAreaCirculo();  
int MostrarPerimetro();
```

- Classe Retângulo:

```
float AreaRetangulo(float l1, float l2);  
int MostrarAreaRetangulo();
```

- Classe Cilindro:

```
int AreaCilindro(float raio, float altura);  
int VolumeCilindro(float raio, float altura);  
int MostraAreaCilindro();  
int MostraVolumeCilindro();
```

Exemplo de Herança Múltipla:

```
class teste:: public class01, public class02{  
    //bloco de comandos  
};
```

Tarefas:

- Implementar um Programa em C++ (Paradigma Orientado a Objetos) de acordo com as classes descritas acima;
- Utilize a Função main descrita abaixo, está não poderá ser modificada;
- Implementar todos os métodos descritos e necessários;
- Instanciar um objeto da classe Cilindro que calcule a sua área e o seu volume. Os cálculos devem observar os métodos herdados e utilizá-los nos cálculos;
- Utilizar "scanf" e "printf" para os valores float, tanto para a entrada como saída de dados.

```
#include<iostream>
#include<conio.h>
#include<stdio.h>
#include "Circulo.h"

using namespace std;

int main() {

    float b,a;
    cilindro cl;

    cout <<"\n\n TESTE DA CLASSE CILINDRO\n";
    cout <<"\n Entre com o raio da base do Cilindro: ";
    scanf("%f",&b);
    cout <<"\n Entre com a altura do Retangulo: ";
    scanf("%f",&a);
    cl.calcular_area_cilindro(b,a);
    cl.calcular_volume_cilindro(b,a);
    cl.mostrar_area_cilindro();
    cl.mostrar_volume_cilindro();

    getch();
}
```