

## **PROGRAMAÇÃO EM C++** com uso do *software Dev C++ (versão 5.1)*

### **ESTUDANTES/AUTORES:**

Washington Nunes da Silva

### **ORIENTAÇÃO:**

Charles Lourenço de Bastos  
Juliana Martins de Bessa Ferreira

### **PROPOSTA:**

#### - Problemática:

É notório a falta de ferramentas de uso fácil para controles de ensaios em concreto estrutural um trabalho ainda executado de modo manual em diversos lugares, visto a situação estou a desenvolver uma pequena amostra da versão 0.b que será comercializada.

#### - Normas de referência:

NBR - 5739:2007 Concreto - Ensaio de compressão de corpos de prova cilíndricos

NBR - 7680:2007

NBR - 9479:2006

NBR - 7500:2004

#### - Livros de referencia:

Informática aplicada em estruturas de concreto armado, PINI.

#### - Matérias Relacionadas:

Física

Resistência Máxima de Compressão

Estatística

Desvio Padrão

Desvio Médio

Imaginando uma situação ideal onde:

- todos os instrumentos estão calibrados em dia e validados
- os traços de concreto estão cadastrados com seus agregados e certificados
- desconsiderando vários detalhes técnicos.

### **CODIFICAÇÃO EM C++:**

```
#include <iostream>
```

```
#include <locale.h>
```

```
#include <stdlib.h>
```

```
#include <math.h>
```

```
#include <conio.c> //biblioteca da borland adaptada para dev c++
```

```
using namespace std;
```



```

    }
    else{
        if((lt>3)&&(lt<=4)){
            d2=2.059;
        }
        else{
            if((lt>4)&&(lt<=5)){
                d2=2.326;
            }
            else{
                if(lt>5){
                    d2=2.534;
                }
            }
        }
    }
}

```

```

// porcentagem de controle usada em calculo estatistico
pc=lt/100;

```

```

//laco para leitura da resistencia
do{
    textcolor(2);
    cout<<"Qual a resistencia esperada em Mpa ? : ";
    textcolor(7);
    cin>>re;
    system("cls");
} while ((0>re)&&(re<100));

```

//laco de leitura de atributo / altura / diametro corpo de prova e aplicacao do coeficiente de correcao h/d

```

for (i=1;i<lt+1;i++){
    textcolor(2);
    cout<<"Corpo de prova numero : ";
    textcolor(7);
    cout<<i<<"\n";
    do{
        textcolor(2);
        cout<<"Digite a Altura";
        textcolor(7);
        cout<<"(em mm)";
        textcolor(2);
        cout<<"do corpo de prova ";
        textcolor(7);
        cin>>h[i];
    } while ((0>h[i])&&(h[i]<100));
    do{
        textcolor(2);
        cout<<"Digite a Diametro";
        textcolor(7);

```

```
cout<<"(em mm)";
textcolor(2);
cout<<" do corpo de prova ";
textcolor(7);
cin>>d[i];
} while ((0>d[i])&&(d[i]<100));

hx=h[i]/d[i];

if(hx<1){
    hd[i]=0.87;
}
else{
    if((hx>1)&&(hx<1.26)){
        hd[i]=0.93;
    }
    else{
        if((hx>1.25)&&(hx<1.51)){
            hd[i]=0.96;
        }
        else{
            if((hx>1.55)&&(hx<1.76)){
                hd[i]=0.98;
            }
            else{
                if(hx<=2){
                    hd[i]=1;
                }
            }
        }
    }
}
}
```

//laco para leitura da força aplicada no rompimento de cada corpo de prova e leitura da força maxima aplicada

```
for (i=1;i<lt+1;i++){
    textcolor(2);
    cout<<"Digite a força aplicada (em Kgf) no corpo de prova nº :";
    textcolor(7);
    cout<<i<<" : ";
    cin>>f[i];
    if(fmax<f[i]){
        fmax<f[i];
    }
    fx=fx+f[i];
}
//desvio padrao
dvp=fx/(d2*lt);
textcolor(2);
cout<<"O desvio Padrao para a amostra e : ";
```

```
textcolor(7);
cout<<dvp<<"\n";

//coeficiente de variacao da amostra
ce=dvp/(fx/lt);
textcolor(2);
cout<<"Coeficiente de Variacao dentro do univeso amostral : ";
textcolor(7);
cout<<ce<<"\n";
textcolor(2);
cout<<"Avaliacao das amostras: ";
if((0<ce)&&(ce<=3)){
    cout<<"Excelente";
}
else{
    textcolor(2);//verde
    if((3<ce)&&(ce<=4)){
        cout<<"Muito Bom";
    }
    else{
        textcolor(2);//verde
        if((4<ce)&&(ce<=5)){
            cout<<"Bom";
        }
        else{
            textcolor(6);//amarelo
            if((5<ce)&&(ce<=6)){
                cout<<"Razoavel";
            }
            else{
                textcolor(4);//vermelho
                cout<<"Deficiente";
            }
        }
    }
}
cout<<"\n";
textcolor(7);
system("pause");
clrscr();
return main();
}
```