

**Ambiente que será utilizado na
prova do dia 18/08/2016**

COMPILAÇÃO

Para compilar um programa em C++ com múltiplos arquivos fonte utilizamos uma linha de comando similar à seguinte:

```
g++ principal.cpp funcao.cpp -o saida
```

O comando anterior compilará o código fonte contido nos arquivos principal.c e funcao.c, os “linkando” e gerando, caso os mesmos não apresente erros de compilação o arquivo executável denominado “saida”.

A execução se dá com ./saida

Linguagem de Programação C++

Exercício:

Crie uma classe Poupanca. Utilize um membro de dados static taxaDeJurosAnual para armazenar a taxa de juros anual para os correntistas. Cada objeto da classe deve conter um membro de dados privado saldo para indicar a quantia que os correntistas têm anualmente em depósito. Forneça uma função membro calculaJurosMensais que calcula os juros mensais multiplicando o saldo pela taxaDeJurosAnual e dividindo por 12; esse valor deve ser adicionado a saldo. Forneça uma função membro static modifiqueTaxaDeJurosAnual que configura taxaDeJurosAnual com um novo valor. Escreva um programa driver para testar a classe Poupanca, instanciando dois objetos diferentes da classe Poupanca, o1 e o2, com saldos de R\$ 2.000,00 e R\$ 3.000,00, respectivamente. Configure a taxaDeJurosAnual como 3%. Em seguida, calcule os juros mensais e imprima os novos saldos de cada um dos correntistas. Então configure a taxaDeJurosAnual como 4%, calcule os juros do próximo mês e imprima os novos saldos para cada um dos poupadores.

Linguagem de Programação Java

COMPILAÇÃO

Para compilar um programa em Java utilizamos o comando:

```
javac nomeDoArquivo.java
```

O comando anterior compilará o código fonte contido no arquivo e gerará, caso o mesmo não apresente erros de compilação, o bytecode armazenando-o no arquivo:

```
nomeDoArquivo.class
```

O qual pode ser executado pela JVM com o seguinte comando:

```
java nomeDoArquivo
```

Linguagem de Programação Java

COMPILAÇÃO

Para compilar uma classe empacotada utilizamos, por exemplo, a linha

```
javac -d . Exemplo.java
```

Vamos ver um exemplo:

```
package edu.univasf.poo;  
public class Horário2  
{  
    private int hora;  
    private int minuto;  
    private int segundo;  
    public Horário2()  
    {  
        this( 0, 0, 0 );  
    }  
    public Horário2( int h, int m, int s )  
    {  
        setHorario( h, m, s );  
    }  
}
```

```

public Horario2( Horario2 horario )
{
    this( horario.getHora(), horario.getMinuto(), horario.getSegundo() );
}
public void setHorario( int h, int m, int s )
{
    hora = ( ( h >= 0 && h < 24 ) ? h : 0 );
    minuto = ( ( m >= 0 && m < 60 ) ? m : 0 );
    segundo = ( ( s >= 0 && s < 60 ) ? s : 0 );
}
public int getHora()
{
    return hora;
}
public int getMinuto()
{
    return minuto;
}
public int getSegundo()
{
    return segundo;
}
public String toString() /**Tópico novo*/
{
    return String.format( "\n%02d:%02d:%02d", hora, minuto, segundo );
}
}

```

Linguagem de Programação Java

Exercício:

Utilizando este conceito implemente uma nova opção no menu do exercício do slide 538 a qual possibilita determinar o custo total apenas dos pacotes que serão entregues à noite.