

UFPB
João Pessoa
Aluno : Daniel de Queiroz Cavalcanti

Assunto: Resposta da Atividade

1) Classe é um modelo, um protótipo a partir do qual os objetos podem ser criados.

Os Objetos são instâncias de uma classe.

Ex: interface em C++ (ou assinatura da classe) é definida por um arquivo de cabeçalho (arquivo.h ou arquivo.hpp)

2} Quando um objeto de uma classe é criado, seus membros podem ser inicializados através de um método construtor. Tais métodos, quando definidos, devem possuir o mesmo nome da classe.

Construtores servem para inicializar dados de um objeto. Uma classe também pode ter vários construtores diferentes implementados. Além disso, eles só podem ser chamados usando o comando **new** ou na declaração de um objeto de uma certa classe.

Ex: CAnimal A1;
CAnimal *A2;
A2 = new CAnimal(); // tanto um como o outro
A2 = new CAnimal; // estão corretos e funcionam

Caso a classe não define um construtor, então o compilador define um construtor *default* sem parâmetros. Assim, ao declarar um objeto, não se coloca parênteses após o nome do mesmo.

No momento que um construtor é definido, o compilador não define mais um construtor default, exigindo do desenvolvedor a declaração deste.

3} os Construtores podem ser chamados usando o comando **new** ou na declaração de um objeto de uma certa classe.

Se o objeto for construído com new (ponteiro), o delete deve ser chamado explicitamente. } Se o objeto for construído sem o new, o destrutor será chamado automaticamente quando o objeto for destruído. }

4} Permite restringir o acesso a variáveis e métodos da classe; Oculta certos detalhes de implementação do usuário da classe; Importante para minimizar os erros de programação; Evita, quebra de integridade dos dados; Facilita a atualização do código.

```

5} class relogio{
private:
int hora;
int minuto;
int segundo;
public: void
setHorario(int h=0, int m=0, int s=0){
if(h<24 && h>0)
hora=h;
else
hora=0;
if(m>60 && m<0)
minuto=00;
else
minuto=m;
if(s<60 && s>0)
segundo=s;
else
segundo=0;
}
int getHorario(int *h, int *m, int *s){
*h=hora;

*m=minuto;
*s=segundo;
}
void avanca(){
segundo++;
if(segundo>59){
segundo=0;
minuto++;
}
if(minuto>59){
minuto=0;
hora++;
}
if(hora>23){
hora=0;
minuto=0;
segundo=0;
}   }};

```

```

#include <iostream>

```

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
#include <string.h>
```

```
using namespace std;
```

```
class relógio
```

```
{
```

```
    private:
```

```
        int hora, minutos, segundos;
```

```
    public:
```

```
        void set_hora(int, int, int);
```

```
        void get_hora(int *, int *, int *);
```

```
        void avanca();
```

```
};
```

```
void relógio::set_hora(int hor, int min, int seg)
```

```
{
```

```
    if(seg>=60)
```

```
    {
```

```
        seg-=60;
```

```
        min+=1;
```

```
    }
```

```
if(min>=60)
```

```
{
```

```
    min-=60;
```

```
    hor+=1;
```

```
}
```

```
if(hor>=24)
```

```
    hor-=24;
```

```
hora=hor;
```

```
minutos=min;
```

```
segundos=seg;
```

```
}
```

```
void relogio::get_hora(int *hor, int *min, int *seg)
```

```
{
```

```
    *hor=hora;
```

```
    *min=minutos;
```

```
    *seg=segundos;
```

```
}
```

```
void relogio::avanca()
```

```
{
```

```
    segundos+=1;
```

```
    if(segundos>=60)
```

```
{
```

```

        segundos-=60;

        minutos+=1;

    }

    if(minutos>=60)

    {

        minutos-=60;

        hora+=1;

    }

    if(hora>=24)

        hora-=24;

}


int main()

{

    relógio rel;

    char op;

    rel.set_hora(0,0,0);

    do

    {

        cout<<"\n\n\t\tHORA(H) \n\t\tMOSTRA(M)\n\t\tAVANÇAR(A)\n\n\t\tOPÇÃO
[ ]\b\b";

        cin>>op;

```

```

switch(op)
{
    case'H': case'h':

        int hora, minuto, segundo;

        cout<<"\n\n\t\tQUE HORAS SAO: "; cin>>hora;

        cout<<"\n\t\tQUE MINUSTOS SAO: "; cin>>minuto;

        cout<<"\n\t\tQUE SEGUNDOS SAO: "; cin>>segundo;

        rel.set_hora(hora, minuto, segundo);

        break;

    case'M': case'm':

        rel.get_hora(&hora, &minuto, &segundo);

        cout<<"\n\n\t\tHora: "<<hora<<":"<<minuto<<":"<<segundo;

        getch();

        break;

    case'A': case'a':

        rel.avanca();

        break;

    default:

        return 0;

        break;

}

}while(1!=0);

}

```

main.cpp

```
int main (){
    relógio relógio1=relógio();
    relógio1.sethora("12");
    relógio1.setminutos("5");
}
```

```
6} class televisao{
private:
int som;
int canal;
int aumentar;
int diminuir;
int n;
public: void
getCanal(int canal=0){
if(canal<10 && canal>0)
canal=1;
else
canal=9;
if(canal>20 && canal<11)
canal=21;
else
canal=10;
```

```
int getVolume(int *som, int *aumentar, int *diminuir){
    *som=aumentar;
```

```
    *som=diminuir;
```

```
int main (){
    canal canal1=canal();
    canal1.setCanal("Canal 10");
    canal1.setSom("Som 30");
}
```

```
7}
```

```
//Classe pessoa:
public class Pessoa{
```

```

private String nome, telefone;
private int idade;

Pessoa(String nome){
    this.nome = nome;
}
Pessoa(String nome, int idade, String telefone){
    this.nome = nome;
    this.idade = idade;
    this.telefone = telefone;
}

public String getNome() {
    return nome;
}
public void setNome(String nome) {
    this.nome = nome;
}

public String getTelefone() {
    return telefone;
}
public void setTelefone(String telefone) {
    this.telefone = telefone;
}

public int getIdade() {
    return idade;
}
public void setIdade(int idade) {
    this.idade = idade;
}
}

```

LETRA B :

```

public class CadastroDePessoas{
    public static void main(String args[]){
        Pessoa pessoa1 = new Pessoa("DANIEL");
        Pessoa pessoa2 = new Pessoa("DAVID", 13, "8823-1033");

        pessoa1.setIdade(pessoa2.getIdade());
        pessoa1.setTelefone("8800-0029");
        pessoa2.setIdade(15);

        System.out.println("exibindo pessoas:\n" + "Nome: " + pessoa1.getNome()

```



```

        + "\nIdade: " + pessoa1.getIdade() + "\nTelefone: " + pessoa1.getTelefone());

        System.out.println("\nexibindo pessoas:\n" + "Nome: " + pessoa2.getNome()
        + "\nIdade: " + pessoa2.getIdade() + "\nTelefone: " + pessoa2.getTelefone());
    }

}

```

```

8) public class Pagamento{
    private double valorPagamento;
    private String nomeDoFuncionario;

    public double getValorPagamento() {
        return valorPagamento;
    }
    public void setValorPagamento(double valorPagamento) {
        this.valorPagamento = valorPagamento;
    }

    public String getNomeDoFuncionario() {
        return nomeDoFuncionario;
    }
    public void setNomeDoFuncionario(String nomeDoFuncionario) {
        this.nomeDoFuncionario = nomeDoFuncionario;
    }
}

```

LETRA B:

```

public class ControleDePagamento{
    private Pagamento pagamentos[] = new Pagamento[2];

    public void setPagamentos(Pagamento pagamentos[]){
        this.pagamentos = pagamentos;
    }

    public double calculaTotalDePagamentos(){
        double total=0;
        for(int i = 0; i < pagamentos.length; i++){
            total+=pagamentos[i].getValorPagamento();
        }
        return total;
    }

    public boolean existePagamentoParaFuncionarios(String funcionario){
        for(int i = 0; i < pagamentos.length; i++){

```

```

        if(funcionario == pagamentos[i].getNomeDoFuncionario()){
            return true;
        }
    }
    return false;
}
}

```

LETRA C :

```

public class SistemaDePagamentos{
    public static void main(String[] args){
        Pagamento arrayPagamentos[] = new Pagamento[2];
        arrayPagamentos[0] = new Pagamento();
        arrayPagamentos[0].setNomeDoFuncionario("DANIEL");
        arrayPagamentos[0].setValorPagamento(15000.00);

        arrayPagamentos[1] = new Pagamento();
        arrayPagamentos[1].setNomeDoFuncionario("LUCIA");
        arrayPagamentos[1].setValorPagamento(21750.00);

        ControleDePagamento pagamentosTotais = new ControleDePagamento();
        pagamentosTotais.setPagamentos(arrayPagamentos);

        System.out.println("Total das despesas: " +
        pagamentosTotais.calculaTotalDePagamentos());
        System.out.println("Pagamento DANIEL? " +
        pagamentosTotais.existePagamentoParaFuncionarios("DANIEL")
        +"\naPagamento MIGUEL? " +
        pagamentosTotais.existePagamentoParaFuncionarios("MIGUEL"));
    }
}

```