



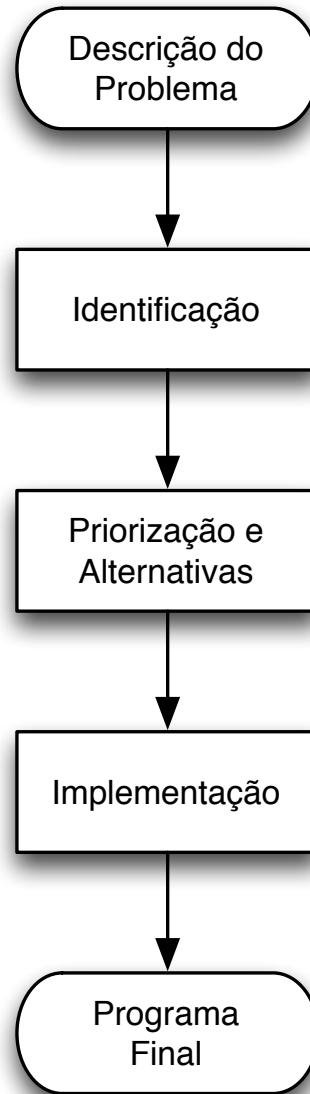
# Programação Orientada a Objetos (TI0142 – 2018.1)

Universidade Federal do Ceará  
Departamento de Computação  
Prof. Lincoln Souza Rocha  
([lincoln@dc.ufc.br](mailto:lincoln@dc.ufc.br))

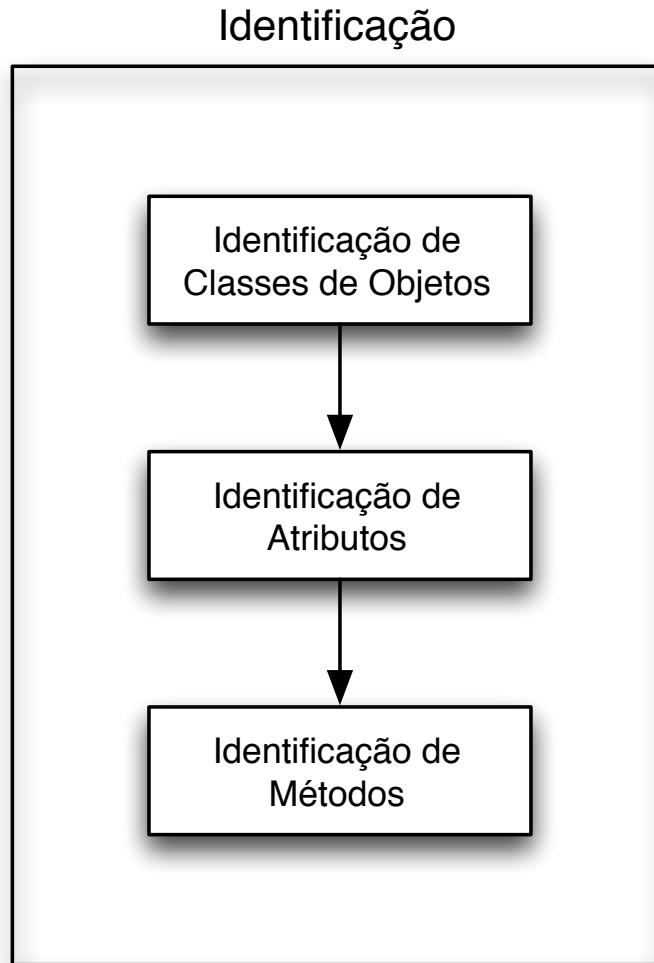
# PROCESSO DE PROGRAMAÇÃO OO



# Visão Geral do Processo



# Fase 1 - Identificação



# Fase 1.1 – Identificação de Classes de Objetos

Classes de Objetos
Classe 1
Classe 2
...
Classe N

# Fase 1.2 – Identificação de Atributos

Classe: Classe1	
Nome do Atributo	Tipo
atributo 1	tipo 1
atributo 2	tipo 2
...	...
atributo N	tipo N

# Fase 1.3 – Identificação de Métodos

Classe: Classe1		
Nome do Método	Tipo de Retorno	Argumentos
método 1	tipo 1	tipo 1 arg 1, ..., tipo n arg n
método 2	tipo 2	tipo 1 arg 1, ..., tipo n arg n
...	...	...
método N	tipo N	tipo 1 arg 1, ..., tipo n arg n

# Fase 2 - Alternativas e Priorização

## Alternativas e Priorização





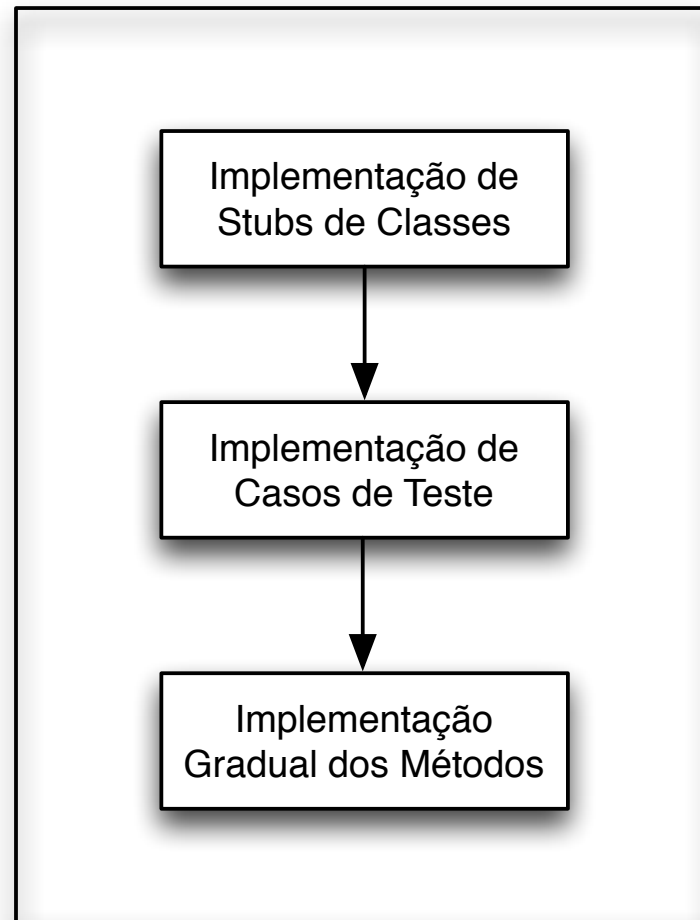
# Fase 2.1 - Alternativas de Implementação de Métodos

ID	Descrição da Representação da Alternativa
$R_1$	Descrição da representação da Alternativa 1
$R_2$	Descrição da representação da Alternativa 1
...	
$R_n$	Descrição da representação da Alternativa 1

Classe: Classe1				
Nome do Método	Alternativa 1	Alternativa 2	...	Alternativa n
método 1	FÁCIL	MÉDIO	...	FÁCIL
método 2	MÉDIO	DIFÍCIL	...	DIFÍCIL
...	...	...	...	...
método N	DIFÍCIL	MÉDIO	...	FÁCIL

# Fase 3 - Implementação

## Implementação



# Fase 3.1 – Implementação de Stubs de Classes

```
01 public class Class1 {  
02  
03     private String atributo1;  
04     (...)  
05     private int atributoN;  
06  
07     public String metodo1() {  
08         return null;  
09     }  
10     (...)  
11     public int metodoN() {  
12         return 0;  
13     }  
14 }
```

# Fase 3.1 – Implementação de Casos de Teste

```
01 public class Class1Teste {  
02  
03     (...)  
04  
05     public void testMetodo1() {  
06         //lógica do teste  
07     }  
08     (...)  
09     public int testeMetodoN() {  
10         //lógica do teste  
11     }  
12 }
```

# Fase 3.2 – Implementação Gradual dos Métodos

```
01 public class Class1 {  
02  
03     private String atributo1;  
04     (...)  
05     private int atributoN;  
06  
07     public String metodo1() {  
08         return this.atributo1;  
09     }  
10     (...)  
11     public int metodoN() {  
12         return 0;  
13     }  
14 }
```

# hands On



# Implementando Datas

Implemente um programa Orientado a Objetos que represente datas em termos de dia, mês e ano. Esse programa deve permitir a data seja incrementada e recuperada para exibição na tela.

# Identificação de Classes de Objetos

## **Classes de Objetos**

Data – utilizada para armazenar e manipular datas



# Identificação de Atributos

Classe: Data	
Nome do Atributo	Tipo
dia	int
mes	int
ano	int

# Identificação de Métodos

Classe: Data		
Nome do Método	Tipo de Retorno	Argumentos
proximaData	void	-
toString	String	-

# Alternativas de Implementação

ID	Descrição da Representação da Alternativa
R <sub>1</sub>	Utilizar três inteiros para representar dia, mês e ano
R <sub>2</sub>	Armazenar o número de dias desde uma data específica (ex. 01/01/0001)

Classe: Data		
Nome do Método	Alternativa 1	Alternativa 2
proximaData	MÉDIO	FÁCIL
toString	FÁCIL	DIFÍCIL

# Implementação de Stub de Classe

```
01 public class Data {
02
03     private int dia; //1 ≤ dia ≤ número de dias do mês
04     private int mes; //1 ≤ mes ≤ 12
05     private int ano;
06
07     public Data(){
08         dia = 07; mes = 04; ano = 2016;
09     }
10
11     public void proximaData() {
12
13     }
14
15     public String toString() {
16         return null;
17     }
18 }
```



# Implementação de Casos de Teste

```
01 public class DataTeste {
02
03     public void testeProximaData() {
04         Data data = new Data();
05         data.proximaData();
06         if(data.toString().equals("8/4/2016")) {
07             System.out.println("testeProximaData passou!");
08         } else {
09             System.out.println("testeProximaData falhou!");
10         }
11     }
12
13     public void testeToString() {
14         Data data = new Data();
15         if(data.toString().equals("7/4/2016")) {
16             System.out.println("testeToString passou!");
17         } else {
18             System.out.println("testeToString falhou!");
19         }
20     }
21 }
```



# Implementação Gradual dos Métodos

```
01 public class Data {
02
03     private int dia; //1 ≤ dia ≤ número de dias do mês
04     private int mes; //1 ≤ mes ≤ 12
05     private int ano;
06
07     public Data(){
08         dia = 07; mes = 04; ano = 2016;
09     }
10
11     public void proximaData() {
12         dia++;
13     }
14
15     public String toString() {
16         return dia + "/" + mes + "/" + ano;
17     }
18 }
```



# Está tudo Ok?



# Implementação Gradual dos Métodos

```
01 public class Data {
02
03     private int dia; //1 ≤ dia ≤ número de dias do mês
04     private int mes; //1 ≤ mes ≤ 12
05     private int ano;
06
07     public Data(){
08         dia = 07; mes = 04; ano = 2016;
09     }
10
11     public void proximaData() {
12         dia++;
13         checkDiaOverflow()
14     }
15
16     private void checkDiaOverflow() {
17         if ( dia > diasDoMes()) {
18             dia = 1;
19         }
20         mes = mes + 1;
21         checkMesOverflow(); } }
22     }
23 (...)
```





# Implementação Gradual dos Métodos

```
24 private checkMesOverflow() {
25     if (mes > 12 ) {
26         mes = 1;
27         ano = ano + 1;
28     }
29 }
30
31 private int diasDoMes() {
32     int[] diasDoMes = {31,28,31,...,31};
33     int result = diasDoMes[mes-1];
34     if(mes == 2 && isAnoBissexto()) {
35         result= result + 1;
36     }
37     return result;
38 }
39
40 private boolean isAnoBissexto() {
41     return (divide(4, ano) && !(divide(100, ano)) || divide(400, ano);
42 }
43
44 private boolean divide(int a, int b) {
45     return b % a == 0;
46 }
47 }
```





# hands On

Tutorial de Programação 03



# Programação Orientada a Objetos (TI0142 – 2018.1)

Universidade Federal do Ceará  
Departamento de Computação  
Prof. Lincoln Souza Rocha  
([lincoln@dc.ufc.br](mailto:lincoln@dc.ufc.br))