

# Programando em Java

## (Programa Principal)

**Mestrado Integrado em Engenharia Informática FCT UNL**

<http://ctp.di.fct.unl.pt/miei/ip/>

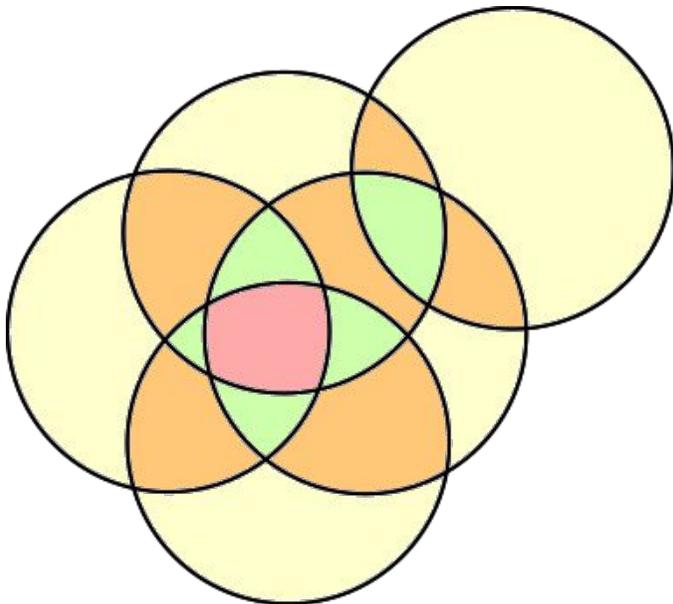
**Corpo Docente 2020/2021**

António Ravara, Artur Miguel Dias, Bernardo Toninho,  
Ema Vieira, Inês Fernandes, Margarida Mamede,  
Miguel Monteiro, Rui Nóbrega

# Problema do Círculo

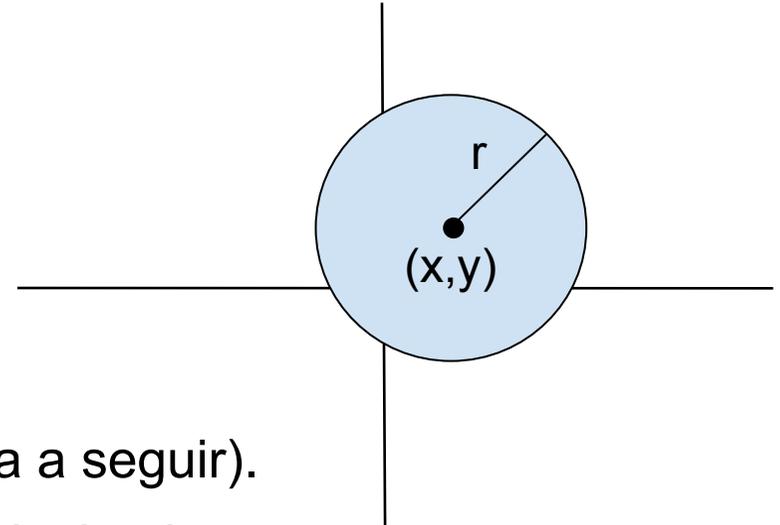
Dado um círculo, definido pelas coordenadas do centro e pelo raio, pretende-se realizar uma das seguintes tarefas:

- calcular o perímetro **ou** a área do círculo;
- saber se um ponto dado pertence ao círculo;
- descobrir a relação entre o círculo e outro círculo dado.



# Problema do Círculo

- O programa lê duas linhas: a primeira define um círculo; a segunda define um **comando**.
- A primeira linha tem três números reais (separados por um espaço), que definem um círculo: a abcissa do centro ( $x$ ), a ordenada do centro ( $y$ ) e o raio ( $r$ ). O raio é um número positivo.
- A segunda linha tem um comando. A linha tem uma das quatro formas seguintes, onde  $x1$ ,  $y1$  e  $r1$  representam números reais:
  - P
  - A
  - LP  $x1$   $y1$
  - LC  $x1$   $y1$   $r1$
- O programa escreve uma única linha (descrita a seguir).
- Todos os números são escritos com 2 casas decimais.



# Comando Perímetro (P)

- (Input) Forma da linha com o **Comando Perímetro**:

P

- (Output) Se  $k$  for o perímetro do círculo, a linha escrita pelo programa tem a forma:

Perimetro:  $k$

- (Exemplo)

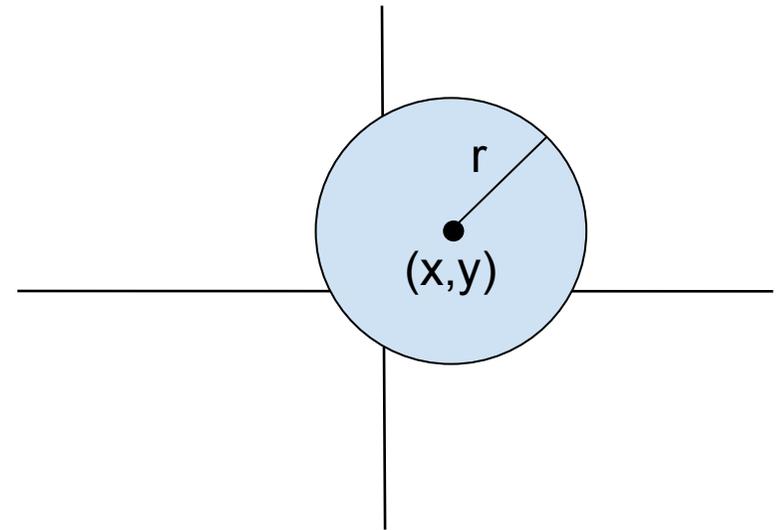
## Input

1.5 2.4 5

P

## Output

Perimetro: 31.42



**Fórmula do perímetro:  $2\pi r$**

# Comando Área (A)

- (Input) Forma da linha com o **Comando Área**:

A

- (Output) Se  $k$  for a área do círculo, a linha escrita pelo programa tem a forma:

Area:  $k$

- (Exemplo)

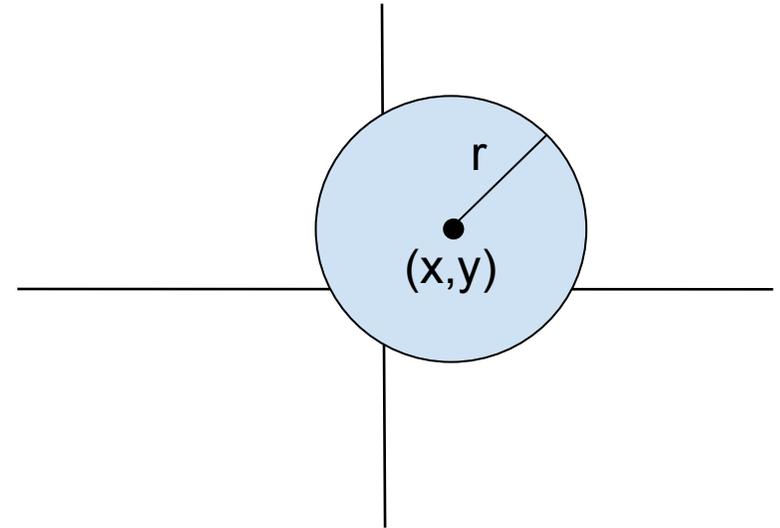
## Input

1.5 2.4 5

A

## Output

Area: 78.54



**Fórmula da área:  $\pi r^2$**

# Comando Localização de Ponto (LP)

- (Input) Forma da linha com o **Comando Localização de Ponto**, onde  $x1$  e  $y1$  são números reais:

LP  $x1$   $y1$

- (Output) A linha escrita pelo programa tem um dos seguintes conteúdos, consoante o ponto  $(x1,y1)$  pertença, ou não pertença, ao círculo:

O ponto pertence ao círculo

O ponto nao pertence ao círculo

- (Exemplos)

## Input 1

1.5 2.4 5

LP 0 0

## Input 2

1.5 2.4 5

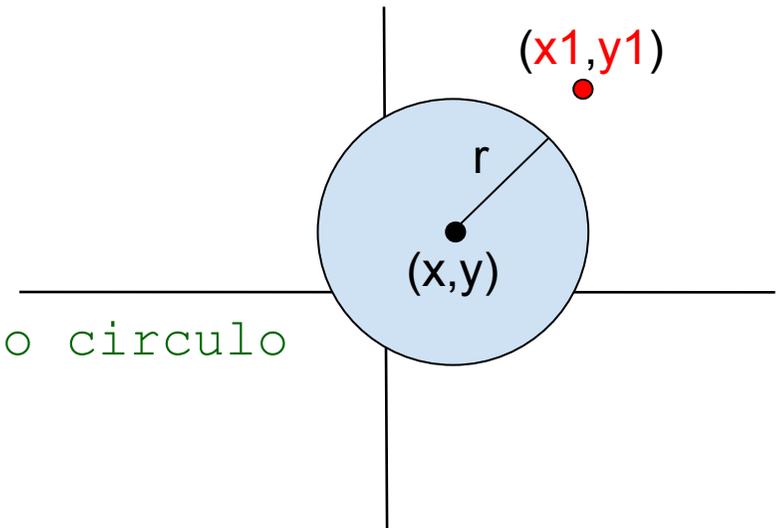
LP -2 -3

## Output 1

O ponto pertence ao círculo

## Output 2

O ponto nao pertence ao círculo



# Comando Localização de Círculo (LC)

- (Input) Forma da linha com o **Comando Localização de Círculo**, onde  $x1$ ,  $y1$  e  $r1$  são números reais:

LC  $x1$   $y1$   $r1$

- (Output) A linha escrita pelo programa tem um dos seguintes conteúdos:

O primeiro círculo contém o segundo

O primeiro círculo intersecta uma parte do segundo

O primeiro círculo não intersecta o segundo

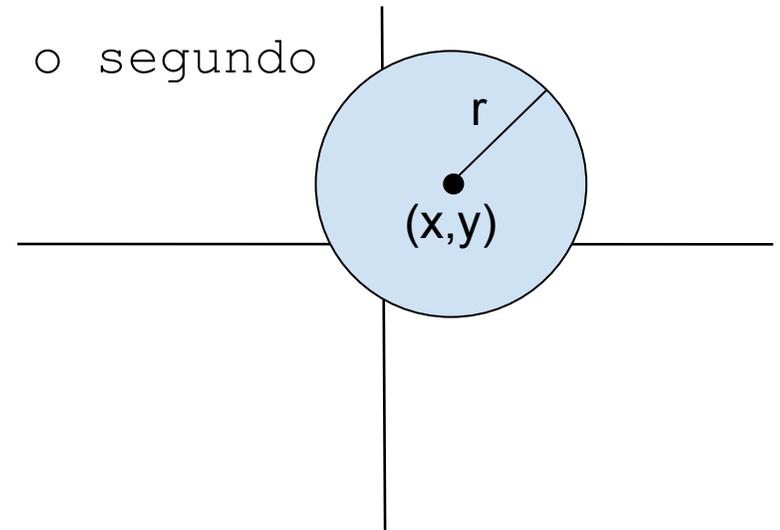
consoante seja a “intersecção” entre

o círculo definido na primeira linha

(com centro em  $(x,y)$  e raio  $r$ )

e o círculo definido no comando,

com centro em  $(x1,y1)$  e raio  $r1$ .



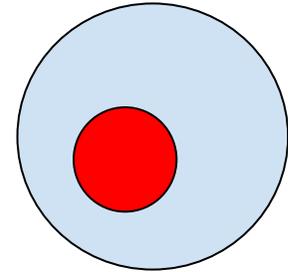
# Comando Localização de Círculo (LC)

## Input 1

```
1.5 2.4 5  
LC 0 0 1
```

## Output 1

O primeiro círculo contém o segundo

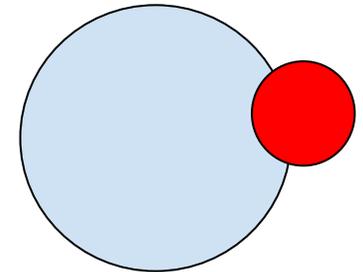


## Input 2

```
1.5 2.4 5  
LC 7 3 1
```

## Output 2

O primeiro círculo intersecta uma parte do segundo

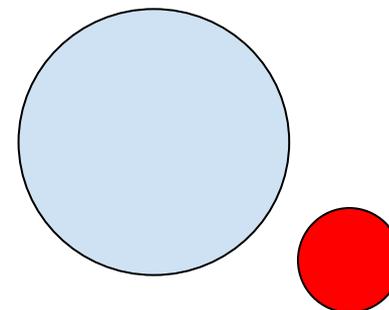


## Input 3

```
1.5 2.4 5  
LC 9 -1 1
```

## Output 3

O primeiro círculo não intersecta o segundo



# Escrita Formatada

```
System.out.printf( formato, v1, v2, ..., vn );
```

escreve a cadeia de caracteres `formato`, “preenchida” com os valores `v1`, `v2`, ..., `vn`, na saída padrão (com  $n \geq 0$ )

## Exemplo:

```
int a = 62123;
```

```
int b = 20;
```

```
System.out.printf("O aluno %d tem %d anos.\n", a, b);
```

escreve:

```
O aluno 62123 tem 20 anos. ↵
```

# Escrita Formatada

O `formato` é uma string com informação sobre o tipo (e a forma) dos valores omissos (a escrever).

- **Alguns tipos frequentes:**

`%d` número inteiro

`%f` número real

`%.nf` número real com  $n$  casas decimais (por exemplo, `%.3f`)

`%s` string

`%c` carácter

- **Alguns caracteres especiais:**

`\n` mudança de linha

`\t` tab

`\"` aspas "

`\\` barra "invertida" \

# Escrita Formatada

```
System.out.printf( formato, v1, v2, ..., vn );
```

escreve a cadeia de caracteres `formato`, “preenchida” com os valores `v1`, `v2`, ..., `vn`, na saída padrão (com  $n \geq 0$ )

## Exemplo:

```
int a = 62123;
```

```
double b = 16.77777777;
```

```
String c = "Adalberto";
```

```
System.out.printf("Vencedor: %s, com o numero %d", c, a);
```

```
System.out.printf("\t(nota %.2f)\n", b);
```

escreve:

```
Vencedor: Adalberto, com o numero 62123 (nota 16.78) 
```

# Plano da Aula

1. Faça o programa considerando que a segunda linha da entrada é um comando **Perímetro** ou um comando **Área**.
  - Quando estiver pronto, avise o professor.
2. Altere o programa para aceitar comandos **Localização de Ponto**.
  - Quando estiver pronto, avise o professor.
3. Altere o programa para aceitar **todos** os comandos.
  - Quando estiver pronto, avise o professor.

**Ninguém pode sair sem ter feito o primeiro programa!**

# Exemplos (P e A)

## Input 1

-2 -3.5 1.74

P

## Output 1

Perimetro: 10.93

## Input 2

-2 -3.5 1.74

A

## Output 2

Area: 9.51

## Input 3

0.5 -7.6 0.0001

P

## Output 3

Perimetro: 0.00

## Input 4

1.1 2.2 123456789

A

## Output 4

Area: 47882831830708840.00

# Exemplos (LP)

## Input 1

-2.2 3.7 4

LP -2.2 -0.3

## Output 1

O ponto pertence ao circulo

## Input 2

-2.8 3.5 0.0001

LP -2.8 3.5

## Output 2

O ponto pertence ao circulo

## Input 3

-2.8 3.5 0.0001

LP -2.80001 3.5

## Output 3

O ponto pertence ao circulo

## Input 4

100 100 100

LP 0 0

## Output 4

O ponto nao pertence ao circulo

# Exemplos (LC)

## Input 1

-2.2 3.7 4

LC -3 2 10

## Input 2

-2.2 3.7 4

LC 2.8 3.7 1

## Input 3

-2.2 3.7 4

LC 0 0 10

## Input 4

-2.2 3.7 4

LC -0.5 0.5 0.1

## Input 5

-2.2 3.7 4

LC -2.2 3.7 4

## Input 6

-2.2 3.7 4

LC 0.5 0.5 0.1

## Outputs 1, 2 e 3

O primeiro circulo intersecta uma parte do segundo

## Outputs 4 e 5

O primeiro circulo contem o segundo

## Output 6

O primeiro circulo nao intersecta o segundo