



**Universidade de São Paulo**

Instituto de Ciências Matemáticas e de Computação

Departamento de Ciências de Computação

SCC0222 – Laboratório de Introdução à Ciência da Computação I

## Exercício: Mapa Logístico

**Professor:** Dr. Marcelo Garcia Manzato ([mmanzato@icmc.usp.br](mailto:mmanzato@icmc.usp.br))  
**Estagiário PAE:** Adam Henrique ([adamh.moreira@gmail.com](mailto:adamh.moreira@gmail.com))  
**Estagiário PAE:** Diego Silva ([diego.fsilva@gmail.com](mailto:diego.fsilva@gmail.com))  
**Monitor:** Bárbara Cortez ([barbara.cortes.souza@usp.br](mailto:barbara.cortes.souza@usp.br))

### 1 Descrição

Implemente um programa em C que produza  $K$  números reais com base em uma equação recorrente chamada **Mapa Logístico**. A equação é dada por:

$$x(t+1) = r \times x(t) \times (1 - x(t))$$

Em que  $t$  é um instante de tempo;  $x(t)$  é um valor real em um instante de tempo  $t$ ; e  $r$  é uma constante definida pelo usuário. O seu programa deve ler da entrada padrão o valor de  $x(0)$  (número real), a constante  $r$  (número real) e a quantidade  $K$  de números a serem produzidos (número inteiro), nesta ordem. A saída deve ser composta por  $K$  linhas que devem conter o instante de tempo  $t$  e o valor  $x(t)$ . Vale ressaltar que, como  $x(0)$  foi dado na entrada, não é necessário imprimí-lo. Para melhor compreender a entrada e saída, observe os exemplos abaixo.

### 2 Instruções Complementares

- Submeta o arquivo .c com seu código no <http://run.codes>

### 3 Exemplos de Entrada e Saída

A seguir são apresentados exemplos de entrada e saída para que você teste seu código enquanto desenvolve o exercício. Este são apenas exemplos ilustrativos, somente uma pequena parte das operações está representada. Enquanto estiver desenvolvendo, elabore novos testes para validar seu código.

#### Entrada

```
0.3 0.8 10
```

#### Saída

```
1 0.168000
2 0.111821
3 0.079454
4 0.058513
5 0.044071
6 0.033703
7 0.026054
8 0.020300
9 0.015910
10 0.012526
```

**Entrada**

0.4 3.0 20

**Saída**

1 0.720000  
2 0.604800  
3 0.717051  
4 0.608667  
5 0.714575  
6 0.611873  
7 0.712453  
8 0.614591  
9 0.710607  
10 0.616934  
11 0.708979  
12 0.618983  
13 0.707529  
14 0.620795  
15 0.706226  
16 0.622413  
17 0.705045  
18 0.623870  
19 0.703969  
20 0.625190