

# LP1 2024/2025 – Projeto Final 2025 - Radar Náutico

Pedro Arroz Serra, Daniel Silveira, Giosuè Muratore, Martijn Kuipers

Deadline: 9/06/2025 @ 23h59hrs no deisi-moodle

25-05-2025 v1.3

---

## 1. Projeto Final 2025 - Radar Náutico

Universidade Lusófona de Humanidades e Tecnologias *Linguagens de Programação I*

Este projeto tem ser desenvolvido na linguagem de programação C. Para além da correta implementação dos requisitos técnicos, serão igualmente avaliados os seguintes critérios:

- O código apresentado deve estar devidamente indentado.
- Os nomes das variáveis terão de ser claros e descritivos, refletindo o conteúdo que armazenam.
- O trabalho é individual e deve ser desenvolvido e submetido como tal.

### 1.1 Objetivo

Desenvolver uma aplicação em C que simule o movimento de barcos, com atualização contínua no tempo. O projeto incide nos seguintes tópicos:

- Ponteiros e gestão dinâmica de memória
- Listas ligadas
- Estruturas compostas
- Leitura e escrita de ficheiros
- Organização modular do código

### 1.2 Requisitos Técnicos

- Utilização obrigatória de listas ligadas para:
  - Lista de frames (BaseDados)
  - Lista de entidades (EntidadeIED)
- É proibido utilizar arrays estáticos para armazenar objetos de natureza dinâmica.
- Não é permitida a utilização de variáveis globais ou estáticas
- O programa não deve ter memory leaks.
- Utilização obrigatória das funções malloc e free.
- O código deve compilar sem erros ou avisos, utilizando o gcc com as seguintes flags: `-Wall -Wextra -g -Wvla -Wpedantic -Wdeclaration-after-statement -lm`
- As seguintes estruturas de dados são de uso obrigatório:

```
1 /**
2  * @struct NoVessel
3  * @brief Representa uma embarcação no jogo.
4  *
5  * Esta contém informação que descreve um elemento no radar.
6  */
7 typedef struct NoVessel {
8     char nome;           /** Identificador do nome do navio */
9     int tipologia;       /** Tipo de embarcação */
10
11     // outros campos podem ser adicionados aqui
12 } NoVessel;
13
14 /**
15  * @struct EntidadeIED
16  * @brief Representa a instância visualizada no radar num determinado frame.
17  *
18  * Armazena a posição, direção de movimento e a referência à embarcação.
19  * ATENÇÃO: Os barcos afundados ou que saíram do radar não devem estar na
20  * lista.
21 */
22 typedef struct EntidadeIED {
23     int posicao[2];       /** Coordenadas (x, y) da posição atual */
24     int velocidade[2];   /** Velocid. horizontal e vertical do barco */
25     NoVessel * no_nautico; /** Ponteiro para a embarcação */
26     struct EntidadeIED *seguinte; /** Ponteiro para a próxima entidade no
27     frame */
28     // outros campos podem ser adicionados aqui
29 } EntidadeIED;
30
31 /**
32  * @struct BaseDados
33  * @brief Estrutura principal da aplicação que armazena os dados do radar.
34  * Responsável por manter o estado atual da simulação, incluindo as
35  * embarcações e a posição visual delas no radar.
36  */
37 typedef struct BaseDados {
38     EntidadeIED * frame_atual; /** Ponteiro para a lista ligada de
39     entidades no frame atual do radar */
40     struct BaseDados *next;    /** Ponteiro para estado do próximo frame */
41     int frame_atual_num;       /** Número do frame atual em segundos */
42     // outros campos podem ser adicionados aqui
43 } BaseDados;
```

Nota: É permitida a adição de campos adicionais em cada estrutura, sendo os apresentados a seguir o mínimo essencial exigido.

**O não cumprimento destes requisitos implica uma penalização de 100%**

### 1.3 Parâmetros de Entrada

Ao executar o programa, o utilizador deve fornecer os seguintes parâmetros:

- Ficheiro de entrada com a lista de barcos (ver secção “Ficheiro de Entrada”)
- Dimensão da grelha, que são dois números (ex: 20x80)

- Número de frames a simular (ex: 1000)
- Ficheiro de saída para guardar o estado da simulação

Exemplo de execução:

```
1 ./radar antes.txt 20x80 1000 depois.txt
```

## 1.4 Formato do Ficheiro de Entrada e Saída

Formato de cada linha:

```
<ID> <LATITUDE> <LONGITUDE> <ÂNGULO> <VELOCIDADE> <TIPO>
```

Exemplo:

```
A 10 20 0 1 2
B 15 25 270 3 4
C 40 20 180 1 2
D 30 25 270 3 4
```

Nota: Os ângulos possíveis são múltiplos de 45 graus, ou seja, 0, 45, 90, 135, 180, 225, 270 e 315 graus. Daí o aluno deve converter os ângulos e velocidade para um vetor de movimento (dx, dy) que representa a direção e velocidade do barco horizontalmente e verticalmente, seguindo os ponteiros do relógio.

Por exemplo:

- A primeira linha é o Barco A, na posição X=10, Y=20, com um ângulo de 0 graus, logo anda para a direita com uma velocidade no sentido dy de 1 casa por segundo.
- O barco B está na posição X=15, Y=25, com um ângulo de 270 graus (para cima) e uma velocidade vy de -3 casas por segundo.
- O barco C está na posição X=40, Y=20, com um ângulo de 180 graus (para a esquerda) e uma velocidade vx de -1 casa por segundo.

A última coluna é o tipo de barco, que é um número entre 1 e 13, detalhado na próxima secção.

## 1.5 Tipos de Barco

Cada barco possui um tipo funcional que define o seu comportamento na simulação.

| Tipo | Nome       | Comportamento/Funcionalidade Especial                             |
|------|------------|---|
| 0    | JonhDoe    | Uma embarcação normal   |
| 1    | ProfPaiMau | Em caso de colisão o barco não afunda                             |
| 2    | Cruzador   | Duplica a velocidade se não houver barcos a 4 casas de distância  |
| 3    | Submarino  | Aparece e desaparece da grelha de 5 em 5 frames                   |
| 4    | Pescador   | Fica parado durante 5 frames de 10 em 10 segundos                 |
| 5    | Traficante | Duplica velocidade se houver outros barcos a 3 casas de distância |
| 6    | Polícia    | Fica parado quando tem um barco a 3 casas de distância            |
| 7    | Turismo    | A velocidade vertical é sempre zero                               |
| 8    | Titanic    | Afunda ao fim de 40 jogadas                                       |
| 9    | Veleiro    | Triplifica velocidade se direção X (horizontal) é positiva        |

|    |             |  |
|----|-------------|--|
| 10 | Rebocador   | Quando a 5 casas de distância de outro barco, a velocidade vertical e horizontal ficam 1 ou -1 de acordo com direção |
| 11 | Cacilheiro  | Ao fim de 10 frames, reduz a velocidade para metade  |
| 12 | PortaAvioes | Navega só na horizontal 1 frame e depois na vertical 1 frame   |
| 13 | Lusitania   | Afunda se esta 3 casas de distancia de outro barco   |

Quando se refere “X casas de distância”, refere-se a barcos que estejam num quadrado X por X com o barco no centro. Ao mudar a velocidade, aplicar o fator ao valor de velocidade vertical e horizontal.

Exemplo:

- “Duplicar velocidade” significa velocidade horizontal vezes 2 e velocidade vertical vezes 2.

## 1.6 Atribuição de Tipos por Número de Aluno

Cada aluno deve implementar apenas os 4 tipos atribuídos de acordo com o último dígito do seu número de aluno.

| Último Dígito | Tipos de Embarcação Permitidos                                |
|---------------|---|
| 0             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 11 (Cacilheiro)  |
| 1             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 4 (Pescador)     |
| 2             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 5 (Traficante)   |
| 3             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 6 (Polícia)      |
| 4             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 7 (Turismo)      |
| 5             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 8 (Titanic)      |
| 6             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 9 (Veleiro)      |
| 7             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 10 (Rebocador)   |
| 8             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 12 (PortaAvioes) |
| 9             | 1 (ProfPaiMau), 2 (Cruzador), 3 (Submarino), 13 (Lusitania)   |

## 1.7 Menu da Simulação

Se o parâmetro da linha de comando contém frames diferente de 0, o programa deve automaticamente avançar a simulação X frames.

### A opção 1 - Atualizar simulação, deve avançar a simulação X frames.

- Pergunta quantos frames quer andar. O tempo é dado em segundos, e cada frame corresponde a 1 segundo.
- O programa deve avançar a simulação o número de frames escrito, atualizando a posição de cada barco.
- Se barcos colidirem, deve remover ambos do radar.
- Após atualizar, grava no ficheiro de saída o resultado da simulação. O nome do ficheiro de saída é o parâmetro da linha de comando.

### A opção 2 - Adicionar um barco novo ou atualizar um barco.

- Deverá perguntar
  - o nome do barco
  - a posição inicial (latitude e longitude)
  - a direção (ângulo)
  - a velocidade.
- O barco é adicionado à lista de barcos.

- Se o barco já existir, deve ser alterado.

#### **A opção 3 - listar barcos que irão colidir no futuro.**

- Só considera colisão quando dois barcos estão na mesma posição.
- Só considera colisões que ocorrem dentro do radar.
- Apresenta coordenada onde vão ocorrer as colisões.

#### **A opção 4 - andar com o tempo para trás X frames.**

- Por exemplo, se temos um histórico com 10 segundos:
  - Deve pedir “Quantos frames deseja voltar?”.
  - Se responder 3, a simulação executa como tivessemos no sétimo frame (10 - 3).

#### **A opção 5 - velocidade média de um barco**

- Calcula o número de casas percorridas horizontalmente (dx) e verticalmente (dy) desde o frame 0 até ao frame atual.
- A distância percorrida é a fórmula “ $\sqrt{dx^2 + dy^2}$ ”, onde “sqrt” é a função raiz quadrada da biblioteca math.h.
- A velocidade média obtém-se da divisão da distância pelo número de frames passados desde o início do programa.

#### **A opção 0 - termina o programa**

- Guarda o estado atual da simulação no ficheiro de saída.
- Se fizemos o que foi dito na opção 4, deve guardar o frame 7 antes de terminar.
- De notar que o ficheiro de saída tem o mesmo formato que o ficheiro de entrada, mas exclui os barcos que afundaram ou saíram do radar.

```
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
```

## **1.8 Submissão e Avaliação**

Submeter até 9 de Junho às 23h59 na plataforma Pandora. O Pandora executa testes automáticos, sugere correções e dá feedback sobre o código. Os testes não contam para a nota. Todos alunos são sujeitos a verificação anti-plágio e avaliação oral obrigatória. A avaliação oral é obrigatória e será feita a todos os alunos, independentemente do resultado dos testes automáticos. A avaliação oral irá decorrer entre 11 e 12 de Junho, e os alunos serão contactados por email para agendar a avaliação.

## **1.9 Cotações**

Todas cotações serão verificadas na avaliação oral. Não existe avaliação automática, apenas avaliação manual.

- 10 valores - Implementação do programa em C, com a utilização de listas ligadas e ponteiros com opção 1, 4 e 0.
- 2 valores - Opção 2
- 2 valores - Opção 3
- 2 valores - Opção 5
- 2 valores - Um tipo de barco
- 2 valores - Todos 4 tipos de barco

- +/- 20 valores - Avaliação oral, **se aluno não demonstra conhecimento do código C chumba projeto.**

## 2. Exemplos

Considerar em todos os exemplos o ficheiro de entrada “antes.txt” dado no início do enunciado e gerar o ficheiro de saída “depois.txt”.

### 2.1 Exemplo de execução do programa, avança 1 frame manualmente via menu

```
./main antes.txt 120x120 0 depois.txt

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 1
Quantos frames deseja avancar? 1
Frame 1 guardado com sucesso em depois.txt
Simulacao atualizada para o frame 1

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 1 guardado com sucesso em depois.txt
A sair do programa...
```

Conteudo do ficheiro de saída “depois.txt” gerado pelo programa:

```
A 11 20 0 1 2
B 15 25 270 3 4
C 39 20 180 1 2
D 30 25 270 3 4
```

## 2.2 Exemplo de execução do programa 5 frames automatico

```
./main antes.txt 120x120 5 depois.txt

Frame 1 guardado com sucesso em depois.txt
Frame 2 guardado com sucesso em depois.txt
Frame 3 guardado com sucesso em depois.txt
Frame 4 guardado com sucesso em depois.txt
Frame 5 guardado com sucesso em depois.txt
Simulacao atualizada para o frame 5

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 5 guardado com sucesso em depois.txt
A sair do programa...
```

Conteudo do ficheiro de saida “depois.txt” gerado pelo programa:

```
A 15 20 0 1 2
B 15 25 270 3 4
C 35 20 180 1 2
D 30 25 270 3 4
```



## 2.3 Exemplo de alterar barco

```
./main antes.txt 120x120 0 depois.txt

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 2

=== Inserir/Alterar Barco ===
Nome do barco (uma letra): B
Posicao inicial (latitude longitude): 69 69
Angulo (multiplo de 45): 180
Velocidade: 99
Tipo do barco (1-13): 1
Barco B alterado com sucesso.
Barco B adicionado com sucesso.

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 0 guardado com sucesso em depois.txt
A sair do programa...
```

Conteudo do ficheiro de saida “depois.txt” gerado pelo programa:

```
B 69 69 180 99 1
D 30 25 270 3 4
C 40 20 180 1 2
A 10 20 0 1 2
```

## 2.4 Exemplo de adicionar novo barco

Conteudo do ficheiro de saida “depois.txt” gerado pelo programa:

```
./main antes.txt 120x120 0 depois.txt

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 2

=== Inserir/Alterar Barco ===
Nome do barco (uma letra): X
Posicao inicial (latitude longitude): 19 75
Angulo (múltiplo de 45): 45
Velocidade: 10
Tipo do barco (1-13): 13
Barco X adicionado com sucesso.

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 0 guardado com sucesso em depois.txt
A sair do programa...
```

Conteudo do ficheiro de saida “depois.txt” gerado pelo programa:

```
X 19 75 45 10 13
D 30 25 270 3 4
C 40 20 180 1 2
B 15 25 270 3 4
A 10 20 0 1 2
```

## 2.5 Exemplo de colisão

```
./main antes.txt 120x120 0 depois.txt

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 3

=== Previsão de Colisões ===
Colisao prevista entre barcos C e A:
  Posicao prevista da colisao: (25,20)

Total de colisões previstas: 1

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 0 guardado com sucesso em depois.txt
A sair do programa...
```

## 2.6 Exemplo de rastrear, avançamos simulação 10 frames e recuamos 5 frames.

```
./main antes.txt 120x120 10 depois.txt
Frame 1 guardado com sucesso em depois.txt
Frame 2 guardado com sucesso em depois.txt
Frame 3 guardado com sucesso em depois.txt
Frame 4 guardado com sucesso em depois.txt
Frame 5 guardado com sucesso em depois.txt
Frame 6 guardado com sucesso em depois.txt
Frame 7 guardado com sucesso em depois.txt
Frame 8 guardado com sucesso em depois.txt
Frame 9 guardado com sucesso em depois.txt
Frame 10 guardado com sucesso em depois.txt
Simulacao atualizada para o frame 10

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 4

=== Rastrear Histprico Reverso ===
Quantos frames deseja voltar? 5

Estado do frame 5:
Barco A: posicao (15,20), velocidade (1,0)
Barco B: posicao (15,25), velocidade (0,-3)
Barco C: posicao (35,20), velocidade (-1,0)
Barco D: posicao (30,25), velocidade (0,-3)

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 5 guardado com sucesso em depois.txt
A sair do programa...
```

Conteudo do ficheiro de saida “depois.txt” gerado pelo programa:

```
A 15 20 0 1 2
B 15 25 270 3 4
C 35 20 180 1 2
D 30 25 270 3 4
```

## 2.7 Exemplo de velocidade média após avançarmos a simulação 3 frames

```
./main antes.txt 120x120 3 depois.txt
Frame 1 guardado com sucesso em depois.txt
Frame 2 guardado com sucesso em depois.txt
Frame 3 guardado com sucesso em depois.txt
Simulacao atualizada para o frame 3

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 5

=== Velocidade Media de um Barco ===
Nome do barco (uma letra): A

Estatisticas do barco A:
  Posicao inicial: (10,20)
  Posicao atual: (13,20)
  Distancia percorrida: 3.00 casas
  Numero de frames: 3
  Velocidade media: 1.00 casas/frame

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 5

=== Velocidade Media de um Barco ===
Nome do barco (uma letra): B

Estatisticas do barco B:
  Posicao inicial: (15,25)
  Posicao atual: (15,25)
  Distancia percorrida: 0.00 casas
  Numero de frames: 3
  Velocidade media: 0.00 casas/frame

=== MENU DA SIMULACAO ===
1. Atualizar simulacao
2. Inserir ou alterar barco
3. Previsao de colisoes
4. Rastrear historico reverso
5. Velocidade media de um barco
0. Sair
Escolha uma opcao: 0
Frame 3 guardado com sucesso em depois.txt
A sair do programa...
```

### 3. Ajuda de visualização

Fica disponível um programa de visualização em Python para ajudar a visualizar o radar.

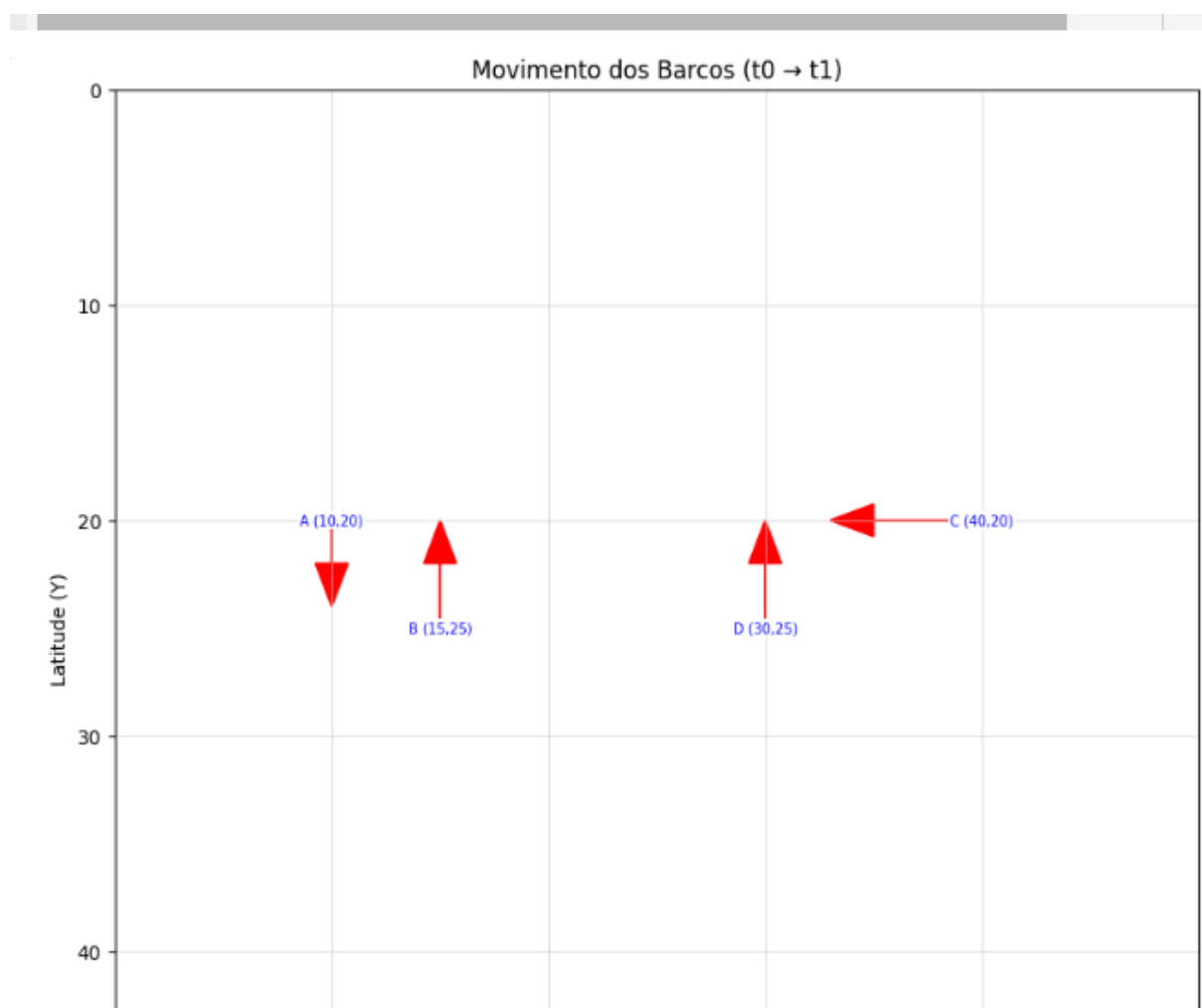
O programa lê o ficheiro de saída e gera uma imagem do radar com os barcos.

Se tiverem conta Gmail, podem usar o Google Colab para correr o programa.

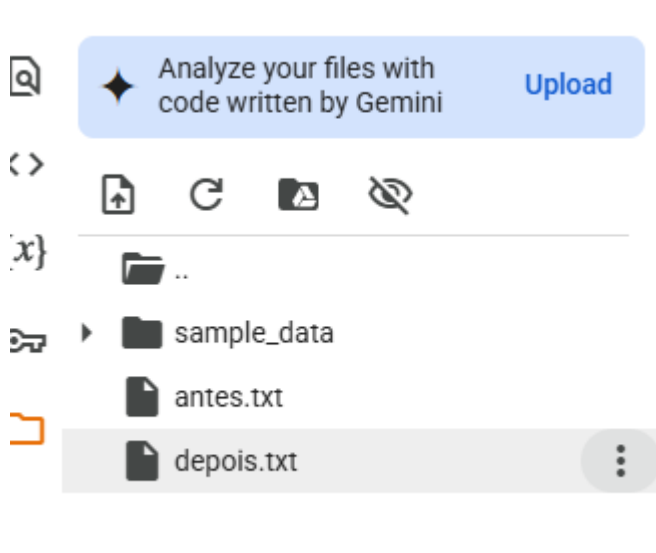
O programa está em modo leitura, mas podem fazer uma cópia do programa para o seu Google Colab e executá-lo a partir daí.

<https://colab.research.google.com/drive/1S7XAGY6dFyjFpCmLQlD6xxUieMN7YYsa?usp=sharing>

Neste programa podem inserir o ficheiro de saída e entrada e verificar o resultado da simulação. O programa gera uma imagem do radar com os barcos. O programa lê o ficheiro de saída e gera uma imagem do radar com os barcos. Segue em baixo um exemplo do resultado, tem os mesmos barcos do exemplo de entrada, mas com a simulação a correr durante 1 segundo. As setas indicam o que os barcos percorreram durante este 1 segundo:



Só precisam de inserir os ficheiros de entrada e saída no local mostrado na imagem abaixo, e correr o programa.



### 3.1 Exemplo barcos normais

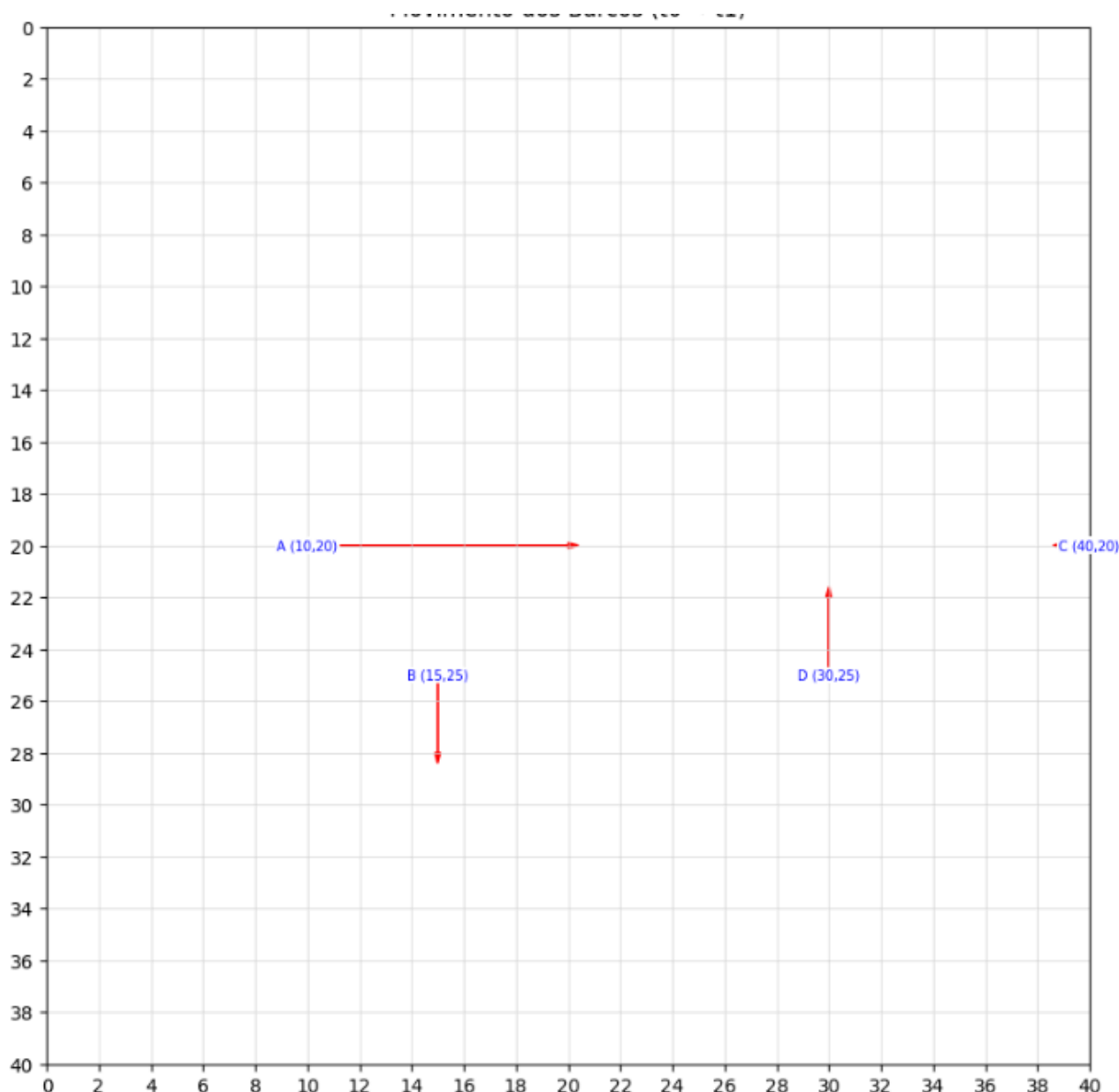
Cada barco move em cada sentido, são todos normais, ou seja, não têm comportamentos especiais.

Antes:

```
A 10 20 0 10 0
B 15 25 90 3 0
C 40 20 180 1 0
D 30 25 270 3 0
```

Depois de 1 segundo:

```
A 20 20 0 10 0
B 15 28 90 3 0
C 39 20 180 1 0
D 30 22 270 3 0
```



### 3.2 Exemplo de barco normais a afundadar

O barco A vai colidir com o barco B após 1 segundo, e ambos afundam.

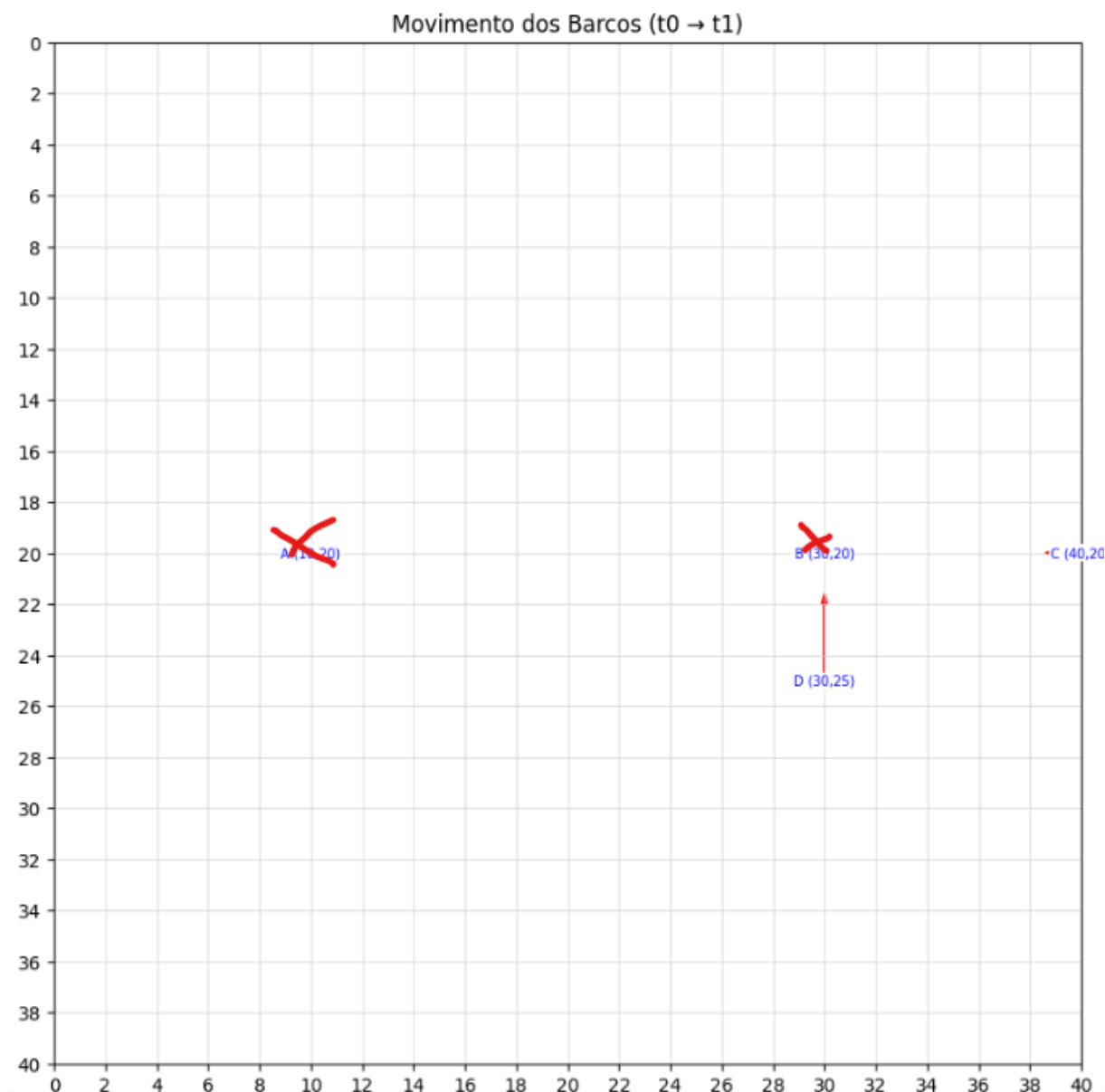
Antes:

```
A 10 20 0 10 0
B 30 20 180 10 0
C 40 20 180 1 0
D 30 25 270 3 0
```

Depois de 1 segundo:

```
C 39 20 180 1 0
D 30 22 270 3 0
```





### 3.3 Exemplo de barco ProfPaiMau

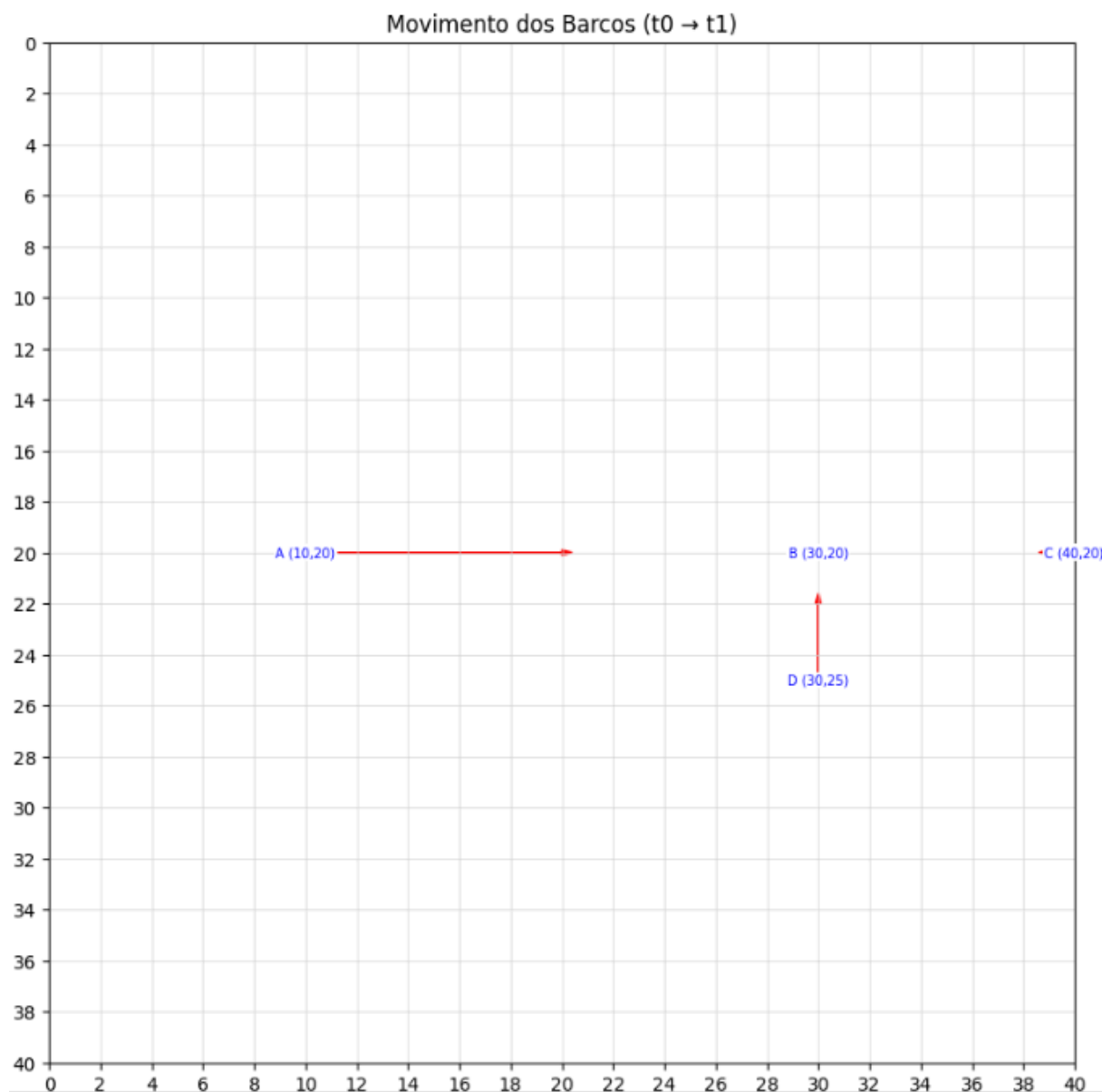
O barco A é um ProfPaiMau, que não afunda ao colidir com o barco B, que é um barco normal.

Antes:

```
A 10 20 0 10 1
B 30 20 180 10 0
C 40 20 180 1 0
D 30 25 270 3 0
```

Depois de 1 segundo:

```
A 20 20 0 10 1
C 39 20 180 1 0
D 30 22 270 3 0
```



### 3.4 Exemplo de barco Cruzador

O barco A é um Cruzador, que duplica a velocidade ao ficar a mais 4 casas de distância de qq barco.

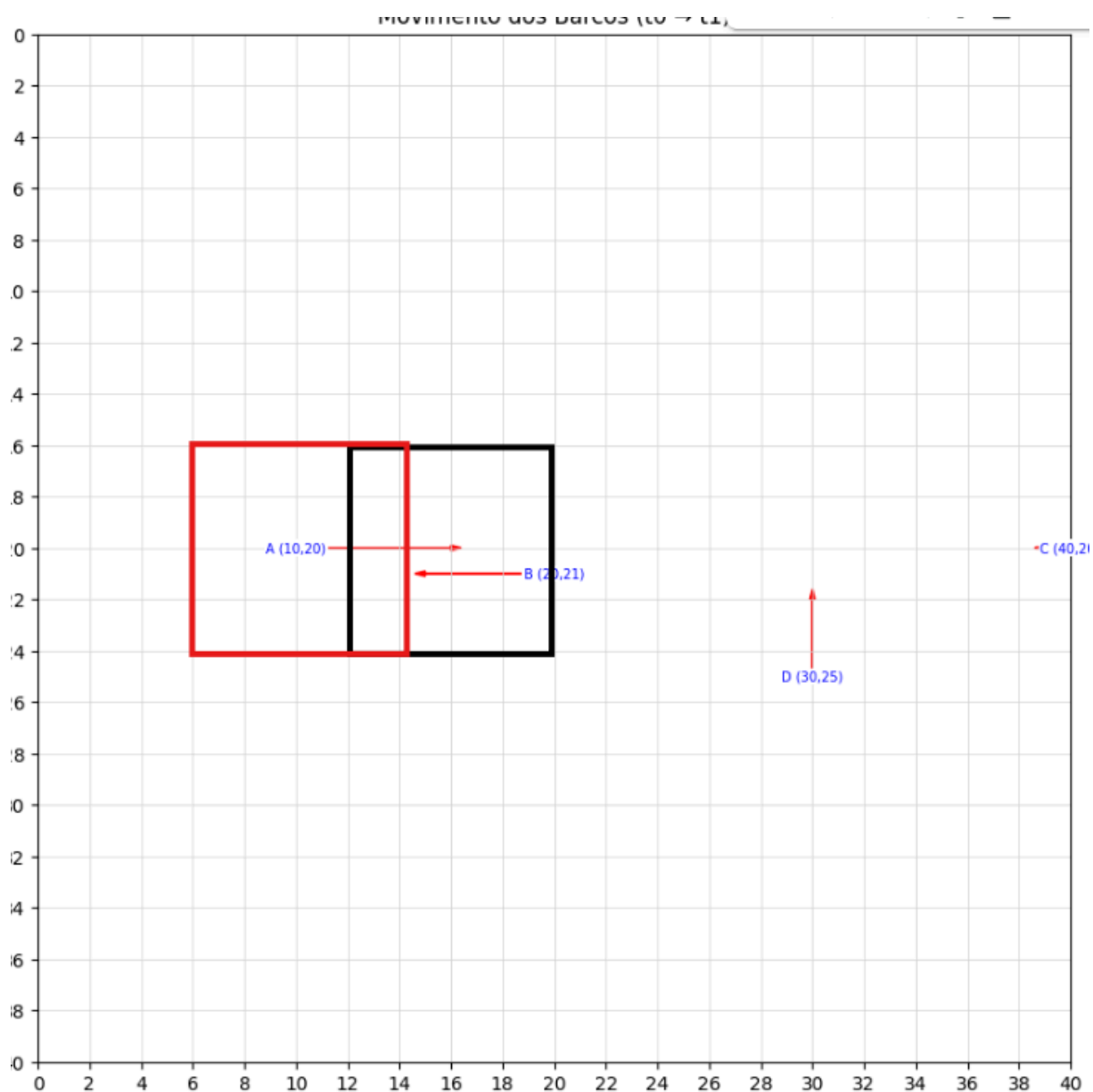
Antes:

```
A 10 20 0 3 2
B 20 21 180 4 0
```

Depois de 1 segundo:

```
A 16 20 0 3 2
B 16 21 180 4 0
```

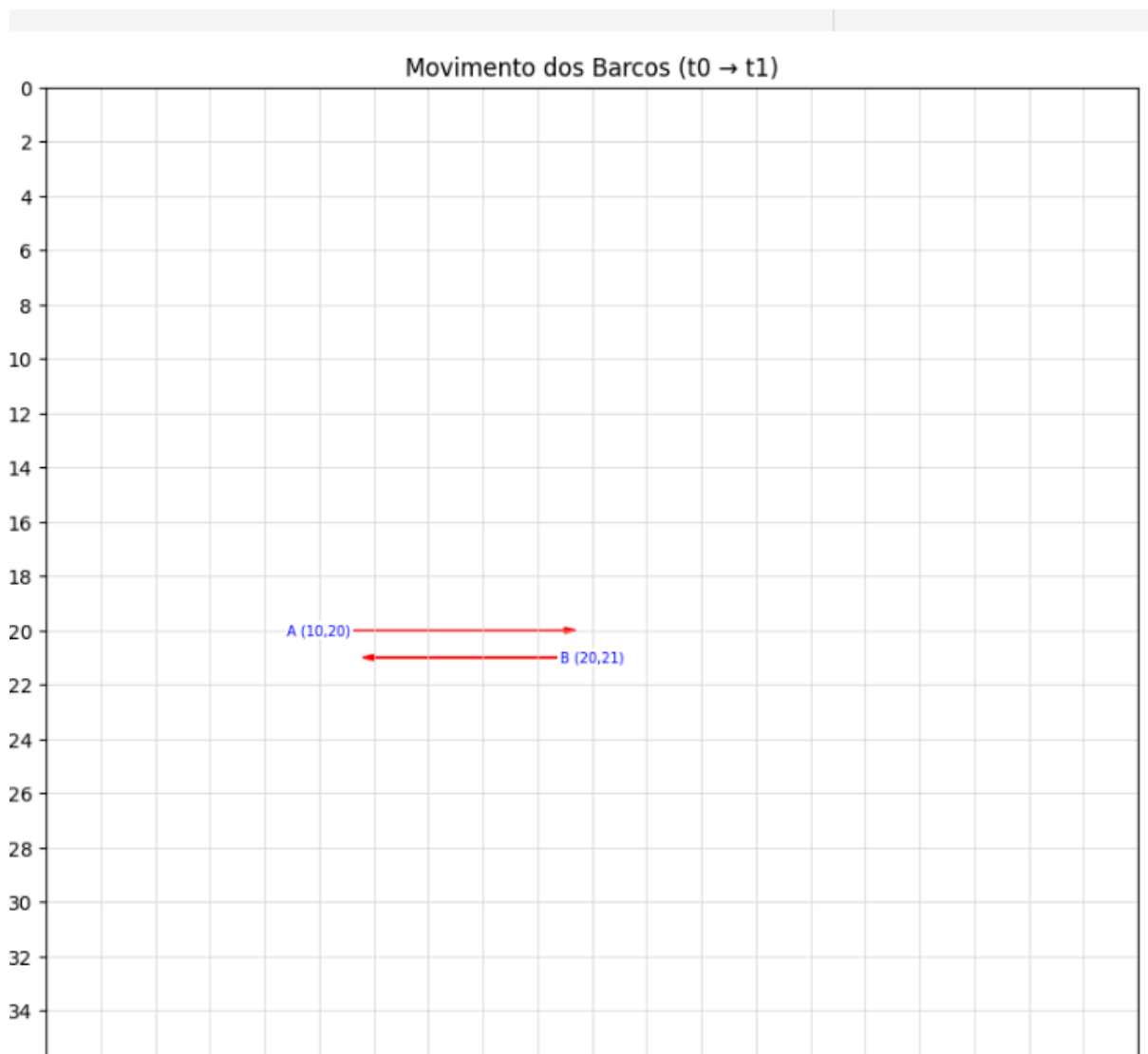
Antes (quadrado preto) o barco B não está a 4 casas de distância do barco A Depois (quadrado vermelho) o barco B está a 4 casas de distância do barco A Logo o barco A reduz velocidade para 3 casas por segundo.



Depois de 2 segundos:

```
B 12 21 180 4 0
A 19 20 0 3 2
```

Reparem que o barco A so anda 3 casas por segundo, porque o barco B está a 4 casas de distância.



### 3.5 Exemplo de barco Submarino

Antes:

```
A 10 20 0 3 2
B 20 21 180 4 0
```

Depois de 4 segundos:

```
B 46 21 180 1 3
A 28 20 0 2 0
```

Depois de 5 segundos (avancei mais 1 segundo):

```
A 30 20 0 2 0
```

Depois de 10 segundos (avancei mais 5 segundos):

```
B 40 21 180 1 3
A 40 20 0 2 0
```

### 3.6 Exemplo de barco Pescador

Antes:

```
A 20 20 0 2 4
B 50 21 180 1 0
```

Apos 5 segundos o barco se mantem parado:

```
A 20 20 0 2 4
B 45 21 180 1 0
```

No segundo seguinte, o barco volta a andar:

```
B 44 21 180 1 0
A 22 20 0 2 4
```

Ele voltara a parar ao fim de 5 segundos.

### 3.7 Exemplo de barco Traficante

Ver exemplo do Cruzador, mas a distancia reduz de 4 para 3 casas de distância.

### 3.8 Exemplo de barco Polícia

Ver exemplo do Cruzador, mas a distancia reduz de 4 para 3 casas de distância e velocidade fica 0 (parado, ver barco parado em Pescador).

### 3.9 Exemplo de barco Turismo

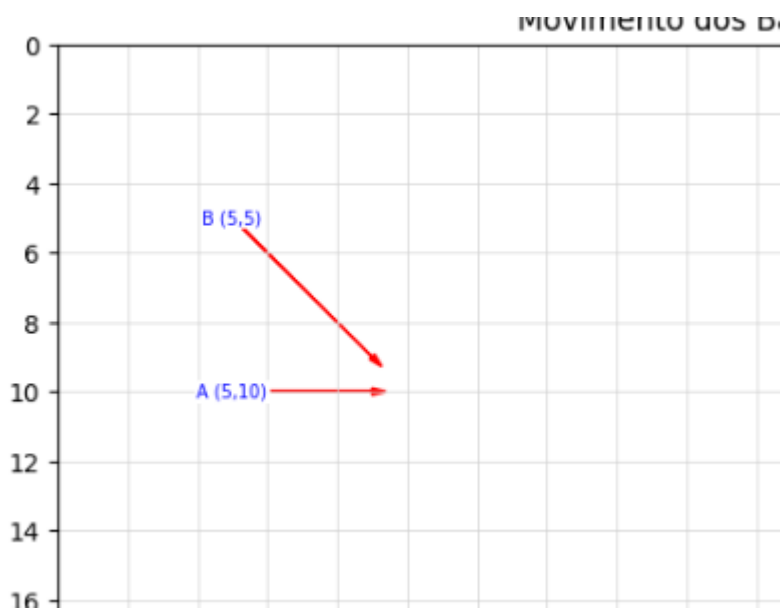
Antes:

```
B 5 5 45 1 0
A 5 10 45 1 7
```

Apos avançar 4 segundos:

```
A 9 10 45 1 7
B 9 9 45 1 0
```

O barco A nao anda na horizontal, apesar de apontar 45 na diagonal igual ao barco B.



### 3.10 Exemplo de barco Titanic

Auto explicativo, remover barco a ao fim de 40 frames.

### 3.11 Exemplo de barco Veleiro

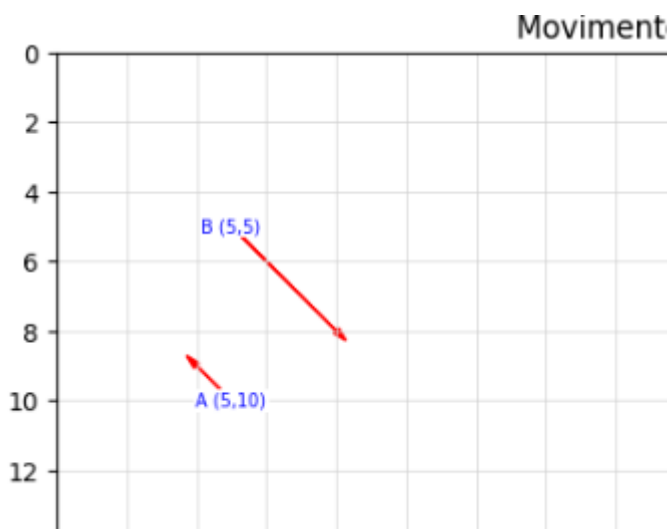
Antes:

```
B 5 5 45 1 9
A 5 10 225 1 9
```

Depois de 1 segundo

```
B 8 8 45 1 9
A 4 9 225 1 99
```

Ambos os barcos andam 1 casa por segundo, mas o barco B triplica velocidade porque a direção X é positiva (anda para a direita).



### 3.12 Exemplo de barco Rebocador

Ver exemplo do Cruzador, mas a distancia sobe de 4 para 5 casas de distância e velocidade fica 1 ou -1 (ver barco parado em Pescador).

### 3.13 Exemplo de barco Cacilheiro

Ver exemplo do Cruzador, mas a velocidade reduz para metade ao fim de 10 frames.

### 3.14 Exemplo de barco PortaAvioes

Inicio, o Barco B é o PortaAvioes

```
B 5 5 45 1 12
A 5 10 225 1 0
```

Apos 1 segundo, so avanca em X.

```
B 6 5 45 1 12
A 4 9 225 1 0
```

Segundo seguinte, so avanca em Y.

```
A 3 8 225 1 0
B 6 6 45 1 12
```

Segundo seguinte, so avanca em X... por ai fora

```
B 7 6 45 1 12
A 2 7 225 1 0
```

### 3.15 Exemplo de barco Lusitania

Semelhante ao exemplo do Cruzador, mas o barco afunda se estiver a 3 casas de distância de outro barco.