

# Banho e Tosa

Professora Patrícia Dockhorn Costa, [patricia.d.costa@ufes.br](mailto:patricia.d.costa@ufes.br)

---



## Descrição

Você foi contratado para desenvolver um sistema que mantém um cadastro simples de gatos e cachorros de uma loja de Banho e Tosa. As seguintes funcionalidades são esperadas:

- Possibilidade de cadastrar gatos e cachorros;
- Organizar animais por nível de agressividade (MANSO ou BRAVO);
- Calcular a renda esperada caso os animais façam o serviço de banho e tosa.

## Arquivos Fornecidos e Esperados

Para facilitar, alguns arquivos da implementação já estão sendo disponibilizados, a saber:

Arquivos	Descrição
<b>Gato.h e Cato.c</b>	Definição e Implementação do Tad Gato
<b>Cachorro.h e Cachorro.c</b>	Definição e Implementação do Tad Cachorro
<b>BanhoTosa.h</b>	Definição do Tad Banho e Tosa
<b>Testador1.c</b>	Testa funcionalidades básicas (vale até 6 pontos)
<b>Testador2.c</b>	Testa funcionalidades intermediárias (vale até 8 pontos)
<b>Testador3.c</b>	Testa todas as funcionalidades (vale até 10 ponto)

Você precisa implementar o Tad BanhoTosa e a lista encadeada necessária para resolver o problema definido (faz parte da solução escolher o tipo de lista adequado). Ou seja, você precisa fornecer 3 arquivos, a saber: **BanhoTosa.c, Lista.h e Lista.c**.

**Importante:**

- **Coloque seu nome em TODOS os arquivos que você produzir.**
- **A lista precisa ser encadeada!**
- **Use tipos opacos, como normalmente fazemos.**

# Dicas

Funções de manipulação de strings <string.h>:

- `strdup`: aloca espaço para string e a copia (`char* destino = strdup(origem)`).
- `strlen`: Retorna o tamanho da string (sem o `'\0'`).
- `strcpy`: Faz a cópia de strings, desde que a string destino tenha tamanho suficiente. `strcpy (destino, origem)`.
- `strcmp`: Compara duas strings. `strcmp (str1, str2)`. Retorna:
  - 0 se str1 for igual a str2.
  - <0 se str1 for menor que str2 (como a sequência das palavras no dicionário)
  - >0 se str1 for maior que str2

Para compilar e gerar o executável

- `gcc -c <seunome>.c` (compila o arquivo e gera arquivo <seunome>.o)
- `gcc -c proglista1.c` (compila o arquivo e gera arquivo proglista1.o)
- `gcc <seunome>.o proglista1.o` (faz a linkagem e gera o executável a.out)