

## 1 Introdução

Seu trabalho consiste em criar um script Bash para gerenciar uma lista de tarefas (*to-do list*). Existem diversos programas que fazem isso de alguma forma, em interfaces web e até mesmo programas que executam no próprio terminal. Algumas ferramentas integram ainda com calendários.

No seu caso, você deve escrever um script Bash chamado `tarefas.sh` que execute no próprio terminal. Você deverá interpretar os comandos do usuário e salvar as tarefas adequadamente.

## 2 Operações do seu programa

Seu programa deve adicionar tarefas que possuem uma descrição. Ao adicionar uma tarefa, seu programa deverá atribuir um número identificador para a tarefa. As tarefas podem, opcionalmente (conforme o desejo do usuário), possuir uma data limite para execução (*deadline*) e uma prioridade.

Deve ser possível listar as tarefas, e escolher a ordenação por identificador, *deadline* ou prioridade. Além disso, seu programa deve permitir modificar uma tarefa, marcar uma tarefa como realizada e também remover uma tarefa.

Se uma tarefa for criada sem data limite, ela não possui data. Se uma tarefa for criada sem prioridade, sua prioridade é a mais baixa possível (1). Ao listar as tarefas, devemos ordenar por identificador a não ser que o usuário selecione outras opções. O usuário pode escolher listar por prioridade e *deadline*. Nesse caso, ordene pelo primeiro critério escolhido e, em caso de empate, ordene conforme o segundo. Em caso de empate, ordene pelo identificador.

Todas as tarefas adicionadas devem ser guardadas em um arquivo chamado `tarefas-todo.txt`. Ao marcar uma tarefa como realizada/feita, você deve mover essa tarefa para um arquivo chamado `tarefas-done.txt` e guardar a data em que ela foi feita. Seu programa deve aceitar que o usuário escolha onde ficam esses arquivos através da variável `TAREFAS_DIR`.

Você pode escolher o melhor jeito de armazenar as informações nos arquivos de texto, desde que o seu programa consiga entender o arquivo e mostrar as informações corretamente.

### 2.1 Exemplos

Abaixo seguem exemplos de funcionamento que o seu programa deve seguir. Os comandos reconhecidos devem ser exatamente conforme mostrados, e as mensagens impressas na tela devem seguir o mesmo padrão mostrado.

```
# adiciona uma tarefa nova
$ tarefas.sh add "Comprar legumes"
```

Tarefa 1: "Comprar legumes" adicionada.

```
# adiciona tarefa com data limite (obrigatório seguir o formato AAAAMMDD)
```

```
$ tarefas.sh add --deadline 20260608 "Consertar máquina de lavar roupa"
```

Tarefa 2: "Consertar máquina de lavar roupa" (20260608) adicionada.

```
# adiciona tarefa com data limite e prioridade (a prioridade padrão é 1)
```

```
$ tarefas.sh add --deadline 20260608 --prio 2 "Limpar mesa"
```

Tarefa 3: "Limpar mesa" (20260608 +2) adicionada.

```
# mostrar todas as tarefas (por padrão, ordena por identificador)
```

```
$ tarefas.sh list
```

```
1: "Comprar legumes"
```

```
2: "Consertar máquina de lavar roupa" (20260608)
```

```
3: "Limpar mesa" (20260608 +2)
```

```
# mostrar todas as tarefas ordenando por data
```

```
$ tarefas.sh list --deadline
```

```
2: "Consertar máquina de lavar roupa" (20260608)
```

```
3: "Limpar mesa" (20260608 +2)
```

```
1: "Comprar legumes"
```

```
# mostrar todas as tarefas ordenando por prioridade
```

```
$ tarefas.sh list --prio
```

```
3: "Limpar mesa" (20260608 +2)
```

```
2: "Consertar máquina de lavar roupa" (20260608)
```

```
1: "Comprar legumes"
```

```
# edita tarefa 1, adicionando prioridade
```

```
$ tarefas.sh edit 1 --prio 3
```

Tarefa 1: "Comprar legumes" (+3) atualizada.

```
# remove tarefa 2
```

```
$ tarefas.sh delete 2
```

Tarefa 2: "Consertar máquina de lavar roupa" (20260608) removida.

```
# completa a tarefa 1
```

```
$ tarefas.sh do 1
```

Tarefa 1: "Comprar legumes" (+3) realizada.

```
# mostra tarefas
```

```
$ tarefas.sh list
```

```
3: "Limpar mesa" (20260608 +2)
```

```
# mostra tarefas concluídas (data e hora no formato AAAAMMDD HHMM)
```

```
$ tarefas.sh list-done
```

```
1: "Comprar legumes" (+3) realizada em 20260601 1645.
```

### 3 Entrega

Você deverá entregar pelo SUAP, até 15/junho, um arquivo *.zip* contendo uma pasta, que por sua vez contém:

- O arquivo `tarefas.sh` do trabalho.
- Um arquivo de texto chamado `README.md` contendo o nome completo dos integrantes do grupo e quaisquer outras informações sobre o progresso do trabalho que julgarem relevantes;

O arquivo compactado descrito acima deve ser nomeado com as iniciais do nome de cada integrante separados por um hífen. Por exemplo: se o Fulano da Silva Sousa fez o trabalho com o João de Souza, o arquivo deve ser nomeado `fss-js.zip`. A pasta, dentro do arquivo compactado, deve ter o mesmo nome, a não ser pela extensão.

O trabalho pode ser feito individualmente ou em duplas.

#### Histórico das Revisões:

- 01/jun/2026 - v1.0: primeira versão.