**Relatório do Projeto de Script Interativo em BASH**

**Descrição Geral do Projeto  
Este projeto é um script interativo em BASH que traz uma série de mini-jogos e simulações para o usuário. A interface principal é um menu, que dá acesso a cada atividade com toques de humor e mensagens divertidas. A ideia é proporcionar uma experiência leve e envolvente, simulando jogos e interações fictícias direto no terminal.**

**➊ Descrição dos Módulos e Sub-rotinas**

* **menu(): Define o menu principal, exibe informações de identificação e apresenta as opções disponíveis para o usuário. Ele calcula o semestre com base no mês atual e apresenta o menu com um cabeçalho decorado. O usuário pode escolher entre cinco opções:  
  • 1) Corrida de Cavalos  
  • 2) Lanchonete  
  • 3) Forca  
  • 4) --Em Obra--  
  • 5) Finalizar o programa.**
* **start(): Inicia o loop principal do programa, chamando a função menu() e aguardando a escolha do usuário. Dependendo da escolha, executa as funções dos mini-jogos ou finaliza o programa.**
* **corrida(): Simula uma corrida de cavalos onde o usuário escolhe seu cavalo e acompanha a movimentação de cada cavalo até que um alcance o final da pista. Se o cavalo escolhido vence, o usuário recebe uma mensagem de vitória, caso contrário, uma mensagem de derrota.**
* **lanchonete(): Simula uma ida a uma lanchonete fictícia, onde o usuário pode escolher itens do cardápio, com um saldo inicial de R$100. Cada escolha reduz o saldo, e o usuário é expulso da lanchonete caso o saldo não seja suficiente.**
* **forca(): Implementa o jogo da forca. Uma palavra aleatória é escolhida e o usuário tenta adivinhar as letras. A cada erro, uma parte do boneco é desenhada até que o número de vidas acabe.**
* **obra(): Simulação que exibe uma série de mensagens animadas relacionadas a uma obra fictícia, com intervalos de espera e telas intermediárias.**
* **c(): Esta função foi adicionada para garantir que o script continue em execução mesmo quando o usuário tenta interromper o programa. Se o usuário pressionar Ctrl+C ou Ctrl+Z, em vez de encerrar o script, o sistema exibe a mensagem "Nã na ni na não", aguarda 2 segundos e reinicia o menu com a função start(). Isso oferece uma experiência mais interativa e divertida, evitando interrupções abruptas no script.**

**➋ Comandos Utilizados e Propósito**

* **#!/bin/bash: Define o interpretador BASH para o script.**
* **clear: Limpa a tela.**
* **date: Obtém data e hora do sistema.**
* **if, else, elif: Estruturas de condição.**
* **while: Cria um loop contínuo até a condição ser atendida.**
* **for: Loop para iterar sobre arrays ou listas.**
* **case: Estrutura de seleção, usada nos menus.**
* **printf: Formata e exibe texto.**
* **echo: Exibe mensagens no terminal.**
* **sleep: Pausa a execução por alguns segundos.**
* **read: Lê entrada do usuário.**
* **declare: Declara variáveis e arrays.**
* **+=: Adiciona valores a uma variável ou array.**
* **${}: Substituição de variáveis.**
* **${#}: Tamanho de uma variável ou array.**
* **sed: Substitui caracteres em strings.**
* **tr: Tradução de caracteres.**
* **seq: Gera uma sequência de números.**
* **trap: Usado para capturar sinais de interrupção do sistema (como Ctrl+C e Ctrl+Z) e redirecionar esses sinais para uma função específica. No caso do código, o trap chama a função c(), que impede o script de ser interrompido e reinicia o menu principal após uma pausa de 2 segundos.**

**➌ Referências Utilizadas**

* **Manual do GNU Bash**: Referência para os principais comandos e sintaxes utilizadas no projeto.
* **Stack Overflow**: Consultas pontuais para uso de arrays associativos, manipulação de strings e formatação de saída.
* **Tutoriais de Scripts Shell (Online)**: Exemplos para entender o controle de fluxo e boas práticas de formatação em scripts BASH.
* **ChatGPT**: Consultas para partes do código que estavam mais difíceis.