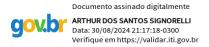
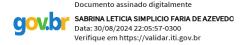
Meta 2: Implementar uma Prova de Conceito de um Programa de Computador (9. CMPLGP1 - Linguagem de Programação)

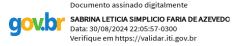
Código utilizado

```
#include <stdio.h>
#include <string.h>
#define MAX ITENS 100
#define DESCRICAO LEN 50
char descricaoEstoque[MAX_ITENS][DESCRICAO_LEN];
int quantidadeEstoque[MAX ITENS];
int numItens = 0;
void adicionarItem(const char* descricao, int quantidade) {
  if (numItens < MAX ITENS) {
     strncpy(descricaoEstoque[numItens], descricao, DESCRICAO LEN - 1);
     descricaoEstoque[numItens][DESCRICAO LEN - 1] = '\0';
     quantidadeEstoque[numItens] = quantidade;
     numltens++;
     printf("Item adicionado com sucesso.\n");
  } else {
     printf("Capacidade maxima de intens alcancados.\n");
  }
}
void atualizarQuantidade(const char* descricao, int quantidade) {
  for (int i = 0; i < numltens; i++) {
     if (strcmp(descricaoEstoque[i], descricao) == 0) {
       quantidadeEstoque[i] = quantidade;
       printf("Quantidade atualizada com sucesso.\n");
       return;
    }
  printf("Item nao encontrado.\n");
```





```
void relatorioEstoque() {
  printf("\nRelatório de Estoque:\n");
  for (int i = 0; i < numltens; i++) {
     printf("Descricao:%s | Quantidade: %d\n", descricaoEstoque[i], quantidadeEstoque[i]);
  }
}
int main() {
  int opcao;
  char descricao[DESCRICAO_LEN];
  int quantidade;
  while (1) {
     printf("\nMenu:\n");
     printf("1. Adicionar Item\n");
     printf("2. Atualizar Quantidade\n");
     printf("3.Relatorio de Estoque\n");
     printf("0. Sair\n");
     printf("Escolha uma opcao:");
     if (scanf("%d", &opcao) != 1) {
        printf("Entrada invalida.Tente novamente.\n");
        while (getchar() != '\n');
        continue;
     getchar();
     if (opcao == 1) {
        printf("Digite a descricao item: ");
        if (fgets(descricao, DESCRICAO LEN, stdin) != NULL) {
          descricao[strcspn(descricao, "\n")] = '\0';
        printf("Digite a quantidade do item: ");
        if (scanf("%d", &quantidade) != 1) {
          printf("Quantidade invalida.Deve ser um numero\n");
          while (getchar() != '\n');
          continue;
        if (quantidade >= 0) {
          adicionarItem(descricao, quantidade);
       } else {
          printf("Quantidade invalida.Deve ser maior ou igual a 0.\n");
        }
```



```
} else if (opcao == 2) {
     printf("Digite a descricao do item: ");
     if (fgets(descricao, DESCRICAO LEN, stdin) != NULL) {
        descricao[strcspn(descricao, "\n")] = '\0';
     }
     printf("Digite a nova quantidade do item: ");
     if (scanf("%d", &quantidade) != 1) {
        printf("Quantidade invalida. Deve ser um numero.\n");
        while (getchar() != '\n');
        continue;
     }
     if (quantidade >= 0) {
        atualizarQuantidade(descricao, quantidade);
     } else {
        printf("Quantidade invalida. Deve ser maior ou igual a 0.\n");
  } else if (opcao == 3) {
     relatorioEstoque();
  } else if (opcao == 0) {
     printf("Saindo...\n");
     return 0;
  } else {
     printf("Opcao invalida. Tente novamente.\n");
}
```

