E-fólio A | Folha de resolução para E-fólio

UNIDADE CURRICULAR: Laboratório de Programação

CÓDIGO: 21178

DOCENTE: Nelson Russo, Pedro Viegas Junior

A preencher pelo estudante

NOME: Vítor Manuel Metrogos Frango

N.º DE ESTUDANTE: 1802925

CURSO: Licenciatura em Engenharia Informática

DATA DE ENTREGA: 27 abr 2024

TRABALHO / RESOLUÇÃO NAS PÁGINAS SEGUINTES:

Este relatório é relativo ao enunciado A da AF2 referente e um programa em linguagem C para gerir o acervo de uma biblioteca.

O código foi escrito em CLion 2023.3.4Build #CL-233.14475.31, built on February 13, 2024 VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o. macOS 14.4.1 e compilado com o GCC

Relativamente à organização do programa e pelos critérios solicitado em enunciado, temos:

a) Módulos e Interface, desempenham funções especificas cada um com uma interface definida pelas suas variáveis e funções sendo também elas disponibilizadas para outros módulos:
    a. Modulo gestão de livros (funções)
        i. Adicionar Livro (**adicionar_livro**)
        ii. Remover Livro (**remover_livro**)
        iii. Editar Livro (**editar_livro**)
        iv. Pesquisar por Livro (**pesquisar_livro**)
        v. Interfaces para outros módulos: protótipos de funções e estruturas de dados (**gestao_livros.h**)
    b. Modulo gestão de empréstimos (funções)
        i. Emprestar Livro (**empresta_livro**)
        ii. Devolver Livro (**devolver_livro**)
        iii. Renovar Empréstimo (**renovar_emprestimo**)
        iv. Carregar empréstimos de arquivo (**carregar_emprestimos**)
        v. Interfaces para outros módulos: protótipos de funções e estruturas de dados e integrações (**gestao_emprestimos.h**)
    c. Modulo Interface (funções)
        i. Exibir menu principal (**exibir_menu_principal**)
        ii. Menu gestão de livros (**menu_gestao_livros**)
        iii. Menu para pesquisa de livros (**exibir_menu_pesquisa_livros**)
        iv. Menu gestão de empréstimos (**menu_gestao_emprestimos**)
    d. Modulo de geração de relatórios
        i. Gerar relatórios de livros emprestados (**gerar_relatorio_emprestimos**)

    Obs: Devido à dificuldade em gerir os ficheiros csv está função não está funcional.

b) Estrutura de dados usado na implementação
    a. Estrutura para livros (Livro) inclui campos para id, título, autor, gênero, numero de copias
    b. Estrutura para empréstimos (Emprestimo) inclui Id do empréstimo, autor, gênero, usuário, copias atuais, copias emprestadas
    c. Estrutura de contagem de usuários (UserCount) mantem a contagem do número de empréstimos
c) Descrição da Função Main (**principal.c)** e funcionalidade global do programa
    a. Inicia o programa carregando os dados necessários e exibindo o menu principal, direcionado após input do utilizador para o modulo **interface.c** onde se podem encontrar a gestão de livros, gestão de empréstimos e gerar relatórios. A mesma grava dados nos ficheiros ***.csv** e finaliza a aplicação

De forma a responder à 2ª parte do EfolioA, para adaptar o código desenvolvido a um programa que faça a gestão de operações de um restaurante, irei avaliar as similaridades e as diferenças entre o código original e o projeto apresentado no enunciado.

a) Modulos/Funções desenvolvidas e que poderei reutilizar
    a. Gestão de dados – Funções para ler dados de um arquivo CSV e armazená-los em memória (reutilizar para carregar pratos e pedidos)
    b. Interface – Funções para exibir menus e interagir com o user podem ser adaptadas facilmente, como é o caso de **exibir_menu_principal.c** e **menu_gestao_livros.c**
b) Codigo que pode ser adaptado e esforço estimado de adaptação
    a. Funções para Adicionar, Remover ou Editar dados – As funções para adicionar, remover, e editar livros podem ser adaptadas para pratos no novo programa, sendo médio o esforço de adaptação, trocar **gestão_livros.c** por **gestão_pratos.c**, e reestruturar as operações
    b. Sistema de relatórios – As funções que geram os relatórios podem ser adaptadas para relatar mais pedidos e mesas ocupadas, o esforço de adaptação sera também médio
    c. Estrutura de dados – a adaptação das estruturas de livros (struct Livro) para pratos é uma tarefa relativamente simples, no entanto será necessário adicionar novas estruturas para pedidos e mesas. Há a necessidade de redefinir campos e tipos de dados
c) Módulos/funções que terei de desenvolver
    a. Gestão de pedidos – modulo que ira controlar pedidos dos clientes, calcular totais e gerir estados dos pedidos
    b. Gestão de mesas – modulo para gerir as mesas incluindo disponibilidade, atribuição de pedidos, controlo de ocupação
    c. Sistema ampliado de relatórios – Tendo em conta a não funcionalidade total do modulo **gera_relatorios.c**, terá de existir uma adaptação grande a este modulo podendo exigir um esforço adicional de forma a agregar os dados
    d. Teste de unidade e integração – apesar de existirem no código inicial, os mesmos terão de ser criados para os novos módulos e a integração dos módulos antigos e novos podem custar um esforço adicional

Em conclusão a adaptação do código existente para o enunciado proposto vai exigir trabalho na reestruturação de partes do sistema e adaptar funcionalidades especialmente para a gestão de pedidos e mesas. Módulos como **interface.c** e **gestão_livros.c** podem ser reutilizados com algumas modificações. Na globalidade o esforço de adaptação sera medio a alto pois há a necessidade de desenvolver novos módulos para funções especificas no sistema do restaurante.

Screenshots dos testes de unidade e de integração