ESTGOH
Escola Superior de Tecnologia e Gestão de Oliveira do Hospital
Instituto Politécnico de Coimbra

Licenciatura Tecnologias e Sistemas de Informação

Programação II

2019/20

Relatório

Ficha 4

Jorge Miguel dos Santos Martins Nº a2019100813

30/04/20

Índice

1 Introdução

Esta foi uma ficha realizada no sentido de compreensão no uso de estruturas, uma nova forma de uso e organização de variáveis.
Após três exercícios de treino encontramos um quarto exercício bastante completo no uso desta forma de organização.
Este relatório será apenas um guia ao código, pois no decorrer da escrita de código foram deixados vários comentários a explicar as funções.

2 Trabalho realizado

2.1 Pergunta 1

Através do código que a professora nos cedeu através da ficha foi pedido para completar esse mesmo código seguindo as instruções deixadas pelos comentários.
No código é possível que deixei esses mesmos comentários e também inseri os meus.
Existem cinco subprogramas neste exercício, inicialmente tenho o "le_data" ao qual como o nome indica serve para receber do utilizador as datas, entretanto inseri algumas condições para que o utilizador não fuja muito ao inserir a data (não inserir caracteres por exemplo), para isso usei um "do while" que facilmente é verificado no código.
Seguidamente temos o subprograma "escreve_data" que como o nome indica serve para que a data seja escrita no stdout do sistema. No subprograma "compara_datas" são realizados alguns "if" como condições para verificar qual a maior data (mais recente) e este subprograma retorna "<" para menor "=" na igualdade e ">" para maior. Este subprograma também é aproveitado pelo subprograma "maior_data" para verificar qual das datas é maior.

2.2 Pergunta 2

Neste exercício tal como aconteceu no primeiro exercício, é dado parte de um código e em seguida nos é pedido para completar o código seguindo algumas instruções deixadas em forma de comentário, mas neste exercício é também pedido em algumas alíneas a realização de outros subprogramas.
O subprograma função verifica a localização do primeiro ponto em relação ao segundo ponto que é dado pelo utilizador. Primeiramente faz uma comparação entre a esquerda e a direita vendo através do "x" (eixo do x) e em seguida faz a comparação se encontra acima ou em baixo através do "y" (eixo do y), estes passos mais especificamente deixei comentado no programa para melhor se perceber a sua funcionalidade.
Seguindo os passos de exercício é de referenciar as condições que usei na declaração do retângulo usando o seu máximo e mínimo tanto no "x" e no "y", isto para que não houvesse erros depois nas medidas da base e da altura, além que um máximo não poderá ser menor que um mínimo.
Para o cálculo da área como deixei também em comentário a explicar, usei o máximo menos o mínimo para saber o tamanho da base como da altura e em seguida apenas apliquei a fórmula.
No último subprograma peço ao utilizador para este indicar um ponto, dando a coordenada "x" e "y" estas são inseridas numa nova estrutura ao qual chamei de "p3" e através da verificação do "if" dentro do subprograma verifico a sua posição.

2.3 Pergunta 3

Neste exercício é solicitado a criação de uma estrutura que retenha os dados de uma fração, para isso criei uma estrutura com os campos numerador e denominador. Primeiramente quando peço ao utilizador para que este insira uma fração não deixo possibilidade que esta tenha denominador 0, em seguida para as frações de soma e subtração, uso a técnica de multiplicação do denominador da fração 1 com o denominador da fração 2 para saber qual será o denominador final e para o numerador multiplico o denominador da fração 1 com o numerador da fração 2 e somo este resultado com o resultado feito através do inverso deste, na subtração o que muda apenas é o facto que no numerador em vez de somar os resultados os subtraio.
Para o caso de multiplicação e divisão não foi preciso usar nenhuma técnica apenas ter atenção ao zero no numerador na divisão que estava a retornar um número diferente de zero ao qual para ultrapassar essa situação usei duas condições que na verificação de o numerador de alguma das funções ser zero então o resultado será zero.
No subprograma potencia antes de qualquer calculo condicionei na verificação se o número elevado for zero então o resultado será 1, seguidamente passei a verificar se existe uma potência negativa que primeiramente troca o numerador e o denominador de uma fração, aqui caso o numerador seja zero fiz com que esta troca não se pudesse realizar e assim apenas escreve um aviso da impossibilidade tal como retorna a estrutura sem qualquer modificação.
Caso o numerador seja viável então após a troca o expoente é multiplicado por -1 para que este passe a positivo e na próxima condição seja dado o resultado da potencia. Nesta condição optei por não usar a função da biblioteca “math.h” mas sim realizar um ciclo ao qual chamo a função multiplicação para que seja feita a multiplicação da fração entre si (resultado deste sempre multiplicado pelo inicial) o número de vezes dado pelo expoente.

2.4 Pergunta 4

Neste exercício mais completo no que toca a organização do programa para com interação com o utilizador decidi que para ser mais viável o seu uso organizei por menu. No decorrer do código é possível se observar vários comentários deixados para explicar para que servem certos comandos. Neste relatório apenas irei focar em alguns pontos principais.
Para que não se ultrapasse a memória do vetor e como é um programa de teste sem uso de ficheiros, inicialmente é pedido ao utilizador a quantidade de funcionários que irá inserir para decidir o tamanho do vetor de estruturas que depois será iniciado.
No subprograma "inserirDados" faço uma condição para verificar se o número já existe e também se na realidade o utilizador está a inserir número, nas strings de nome e tarefa acabei por não realizar alguma condição pois caso algum engano é possível mudar o nome do funcionário ou tarefa em outro subprograma, mas o número é fixo.
No subprograma de "actualizaNum" que pesquisa um funcionário através do número para os seus dados serem actualizados, decidi criar um submenu para que o utilizador não tenha que passar em todos os campos obrigatoriamente e assim apenas atualizar o que deseja.
Para o subprograma "ordemCrescente" adaptei o algoritmo de "Insert Selection" já utilizado em outra ficha para ordenar os funcionários por ordem crescente em relação ao número. No caso do próximo subprograma "ordemAlpha" primeiramente tentei usar o algoritmo "Insert Selection" mas estava a ocorrer um erro na organização do primeiro funcionário então decidi usar o "Bubble Sort" também adaptado de uma ficha anterior para que os funcionários sejam ordenados por ordem alfabética. No "Bubble Sort" não usei a condição que tinha inserido numa ficha anterior para deixar este algoritmo na sua forma original.
Além dos subprogramas "listaFuncionarios" que lista todos os funcionários e "lista500euros" que funciona como o subprograma anterior, mas com a condição que apenas escreve no "stdout" os funcionários com salário acima dos 500 euros, neste relatório penso que seja importante referenciar o porquê de existir um subprograma "listaPorNome" e um outro "procuraPorNome". Inicialmente na minha interpretação pensei que seria pedido para escrever um algoritmo que escrevesse em tabela todos os funcionários com o nome dado pelo utilizador, mas foi-me informado que o subprograma deveria devolver uma estrutura que seria o primeiro funcionário a ser encontrado com o nome igual (partindo do prossuposto que não existam funcionários com nome igual), então dessa forma realizei o subprograma "procuraPorNome" que caso o utilizador apenas ponha o nome incompleto de um funcionário na primeira condição de igualdade na lista ele devolve essa igualdade.
Neste relatório não estou a entrar em grandes pormenores na explicação de código pois deixei esse contexto no código no formato de comentário.

3 Conclusão

Na realização desta ficha de trabalho foram obtidos conhecimentos importantes no manuseamento de estruturas, estes servirão de conhecimento base para a aprendizagem do uso de listas ligadas.
Este relatório foi realizado apenas como guia de contextualização do código estando em sintonia com o que foi dado em forma de comentários para uma melhor perceção das decisões e logicas por detrás do que foi escrito.

4 Referências

Listar consultas usadas

- Luis Damas Liguagem C

5 Anexos

Exer1.c

Exer2.c

Exer3.c

Exer4.c