

AED1 - Preparação 01

Tema: Armazenamento secundário e grupos de dados homogêneos e heterogêneos

Exercícios:

Usar o modelo de lista de métodos para desenvolver e testar programas em linguagem C para atender aos enunciados que se seguem.

```
/*
Funcao principal.
@return codigo de encerramento
@param argc - quantidade de parametros na linha de comandos
@param argv - arranjo com o grupo de parametros na linha de comandos
*/
int main ( int argc, char* argv [ ] )
{
    // definir dado
    int opcao = 0

    // identificar
    printf ( "%s\n", "Exemplo0100 - Programa = v0.0" );
    printf ( "%s\n", "Autor: _____" );
    printf ( "\n" );           // mudar de linha

    // acoes

    // repetir
    do
    {
        // para mostrar opcoes
        printf ( "\n%s\n", "Opcoes:" );
        printf ( "\n%s" , "0 - Terminar" );
        printf ( "\n%s" , "1 - Method_01" );
        printf ( "\n%s" , "2 - Method_02" );
        printf ( "\n%s" , "3 - Method_03" );
        printf ( "\n%s" , "4 - Method_04" );
        printf ( "\n%s" , "5 - Method_05" );
        printf ( "\n" );

        // ler a opcao do teclado
        printf ( "\n%s", "Opcao = " );
        scanf ( "%d", &opcao );
        getchar();           // para limpar a entrada de dados

        // para mostrar a opcao lida
        printf ( "\n%s%d", "Opcao = ", opcao );
    }
```

```

// escolher acao dependente da opcao
switch ( opcao )
{
    case 0: /* nao fazer nada */ break;
    case 1: method_01 ( ); break;
    case 2: method_02 ( ); break;
    case 3: method_03 ( ); break;
    case 4: method_04 ( ); break;
    case 5: method_05 ( ); break;
    default: // comportamento padrao
        printf ( "\nERRO: Opcao invalida.\n" );
        break;
} // end switch
}
while ( opcao != 0 );

// encerrar
printf ( "\n\nApertar ENTER para terminar." );
getchar( ); // aguardar por ENTER
return ( 0 ); // voltar ao SO (sem erros)
} // end main ( )

/*
----- documentacao complementar

----- notas / observacoes / comentarios

----- previsao de testes

a.) a
b.) 4.2
c.) 10
d.) 1
e.) abc def
f.) abc def

----- historico

Versao    Data        Modificacao
0.1       __/___        esboco

----- testes

Versao    Teste
0.1       01. ( OK )    identificacao de programa

*/

```

Exercícios:

DICAS GERAIS: Consultar os Anexos C 01 e C 02 ou na apostila o capítulo 05 para outros exemplos.
Prever, testar e registrar todos os dados e os resultados obtidos.

01.) Definir e testar um procedimento (01) para:

- definir uma estrutura de dados para guardar um par de valores inteiros;
- ler pares de valores reais de arquivo (DADOS_1.TXT);
- parar quando houver um par for igual a (0,0), ou quando acabarem os dados no arquivo;
- se um par corresponder às coordenadas de um ponto,
- calcular a distância entre dois valores consecutivos e
- gravar cada valor calculado no arquivo (SAIDA_1.TXT)

Usar constantes reais (**double**) em expressões que envolvam valores reais (**double**).

Exemplo: (5,2) e (0,0) -> distância = raiz (quadrado(5-0)+quadrado(2-0));.

02.) Definir e testar um procedimento (02) para:

- ler o arquivo (SAIDA_1.TXT) gerado no exercício anterior e armazenar as distâncias em arranjo;
- calcular a maior e a menor distância entre elas usando funções
- que recebam o arranjo e seu tamanho como parâmetros.

Exemplo: { 5, 2.0, 1,0, 5.0, 4.0, 3.0 }

03.) Definir e testar um procedimento (03) para:

- ler o arquivo (SAIDA_1.TXT) gerado no primeiro exercício e armazenar as distâncias em arranjo;
- calcular a distância média usando uma função
- que receba o arranjo e seu tamanho como parâmetros.
- A média deve ser calculada descartando a maior e a menor distância.

Exemplo: { 5, 2.0, 1,0, 5.0, 4.0, 3.0 }

04.) Definir e testar um procedimento (04) para:

- ler pares de valores reais de arquivo (DADOS_1.txt) e
- armazenar cada primeiro valor em uma coluna de matriz e o segundo na coluna seguinte;
- comparar sucessivamente os valores na primeira coluna e
- trocar os conteúdos de linhas, até que a primeira coluna
- esteja ordenada de forma crescente.
- Se houver valores iguais, desempatar pelo segundo valor.

DICA: Usar mais de uma repetição para fazer os testes
de linhas consecutivas mais de uma vez.

Exemplo: sequência = { (5,2), (2,3), (3,1), (5,1), (1,4), (0,0) }

05.) Definir e testar um procedimento (05) para refazer o exercício (04) para

- definir e usar uma função que receba a matriz de pares ordenada
- como parâmetro, bem como a quantidade de linhas e colunas (2) e
- calcular a distância entre o primeiro par e o último par ordenados.