|  |  |
| --- | --- |
| https://aoquadrado.catracalivre.com.br/wp-content/uploads/sites/4/2014/07/programacao-em-c.jpeg  Manual de Utilização  LP 1 | LESI-PL  O presente documento vai descrever como a aplicação de consola funciona e resolve os problemas da ficha prática.  Joaquim Cardoso  A10201 |

**Introdução**

Para a realização do trabalho prático foi pedido um documento para demonstrar ao utilizador como a aplicação da consola funciona e como foram resolvidos os problemas, e de que forma foram criados e impostos os algoritmos.

Passo de seguida a expor os problemas solucionados na aplicação de consola.

**Problemas**

1. **Arrays (**Obrigatório
2. Determina a ocorrência de um valor num dado array.
3. Verifica se determinado valor existe num array se existir da posição ,senão, devolve falso!
4. Devolve os conjuntos de valores que correspondem aos que são maiores e outro aos que são menores que determinado.
5. **Polinómio de 2º grau**
6. Fórmula resolvente.
7. Derivada de f(x).
8. **Funções Matemáticas**
9. Fatorial
10. Potência
11. Somatório
12. **Funções Estatísticas**
13. Média Aritmética
14. Moda
15. Mediana
16. Desvio Padrão
17. O Maior valor
18. O Menor valor
19. **Conversor de Unidades**
20. Unidades de Comprimento
21. Unidades de Energia
22. Unidades de Temperatura
23. Unidades de Velocidade
24. **Páginas Amarelas**
25. **Gestão da Solução** **(**Obrigatório**)**

**Solução**

Irei falar primeiro como é constituída a aplicação.

Esta aplicação pode ser ver como uma solução com dois projetos, um deles é onde se verifica a existência de todas as fórmulas que resolvem os problemas pedidos, ou seja algoritmos. Este código está guardado numa DLL, uma biblioteca dinâmica, que precisa de ser adicionada ao código principal.

O outro código é considerado por programadores o Main, principal, e onde é verificada uma programação simples com objetivo de chamar outros métodos com soluções a problemas.

Na aplicação apliquei um sistema de menus, pois sempre é possível voltar atras ou sair da aplicação após executar qualquer ação. Tambem apliquei um sistema de TryParse , ou seja a aplicação só vai ler os valores que redirecionem para alguma opção identificada se nunca colocar uma opção correta a aplicação não vai sair do menu inicial.

Dada a apresentação da estrutura do código irei demonstrar como foram resolvidos os problemas propostos para este trabalho prático e apresentada a solução ao utilizador.  
  
Em primeiro lugar foi feito um menu pedido no ponto 7. Como o menu é a primeira interação como utilizador, foi por onde comecei. Criei então um menu que espera que o utilizador introduza um numero inteiro para escolher uma das sete opções, se não introduzir um inteiro não tem mal nenhum o código não irá dar erro, irá voltar a perguntar, dependendo dessa opção o utilizador vai aceder a diferentes conteúdos, pode sempre voltar atrás na aplicação ou sair quando apresentado um menu , após inicializar a aplicação, esta mesma aplicação pode ser fechada com a introdução do valor 0 seguida por um enter.

Após o menu concluído foi criada a solução do 1º problema, a determinação da ocorrência de um valor no array. Primeiro foi preciso criar um array com n valores aleatórios e de seguida uma variável para ser um valor que irá ser procurado ao longo do array, mas antes de tudo para a própria verificação do utilizador os valores do array iram ser escritos no ecrã para o utilizador poder conferir se o resultado da solução dos problemas. Então só após disso é feita a comparação com cada valor pertencente no array com o valor predefinido, qualquer que seja o valor correspondente a essa ação ele vai ser enviado para o ecrã, tal como o número que vai ser procurado no array.

Para a verificação de determinado valor no array, e se verdadeira então enviar a posição, é feita a comparação desse mesmo valor com os valores existentes no array, após disso o programa vai verificar se isso é verdadeiro, se for verdadeiro vai verificar em que posição está esse número procurado, mesmo que o número procurado esteja a ocorrer n vezes no array, o numero procurado em todas as posições onde ele se encontrar no array, vai ser mostrado ao utilizador. Se não for verdadeira a existência do valor procurado no array, vai ser respondido False.

Para o final do ponto obrigatório é feita a verificação de quantos são maiores que o valor n e quantos são menores que valor n e a descrição de ambos os valores, os que são maiores que n e os que são menores que n. Para fazer a verificação é percorrido o array e por cada número maior ou menor que n incrementamos o valor de uma variável. Para mostrarmos ao utilizador quem são os maiores e menores, criamos uma array para ambos, menores e maiores que n, e fiz a verificação á array que contem o conjunto de valores. Se um numero pertencente á array onde contem o conjunto de valores for menor ou maior que n , então esse numero é guardado numa outra array de números maiores que n ou menores que n. De seguida da verificação de cada elemento do conjunto de valores é mostrada a array dos números maiores e menores. Se um array estiver vazia não irá mostrar qualquer valor.

No ponto dois é pedida a criação de um algoritmo para descobrir dois pontos utilizando a fórmula resolvente. Então tendo numero a, b e c já predefinidos é aplicada a fórmula resolvente. Se A for zero e/ou delta menor que zero então, não é aplicada a fórmula resolvente e é apresentada uma mensagem a definir o erro.

Para a derivada são aproveitados os mesmos números usados para calcular a fórmula resolvente mais o valor de x, com isso aplicamos a fórmula da derivada para equações de grau dois e apresentamos o resultado ao utilizador. Tal como a solução do ponto numero um, é demonstrado o valor das variáveis usadas.

De acordo com o ponto número três é pedido para aplicar uma fórmula para o somatório, fatorial e potência. Foi aplicada a fórmula em parâmetros predefinidos e o resultado é mostrado ao utilizador.

No ponto quatro é pedido a solução a diversos problemas sendo estes: média aritmética, moda, mediana, desvio padrão, o maior valor e o menor valor.

Para a média aritmética é verificada a dimensão do conjunto de valores, e esses próprios valores são somado e divididos pela dimensão do seu conjunto fazendo assim a média aritmética.

Para moda é dado um conjunto de valor já predefinido e esse conjunto é ordenado de forma crescente, após essa ordenação é verificado quantas vezes ocorre cada número e é colocado esse valor num conjunto de valores, após saber-se quantas vezes cada valor ocorre é feita outra ordenação mas desta vez de forma decrescente para saber-se qual é o valor que ocorre mais vezes, se existir mais que um valor são ambos enviados para um array que irá ser apresentado na consola.

Para a mediana foi aplica a fórmula matemática simples, primeiro os valores foram ordenados e de seguida verificamos se o número total de valores é par ou impar e então após esse passo determina-se a mediada.

Para o desvio padrão foram aplicadas apenas fórmulas matemáticas.

Para o maior valor e menor valor, foi verificado cada número e o maior ou menor deles era guardado e mostrado ao utilizador.

Para o ponto 5 foram criados dois menus, uma para a escolha do tipo de unidades que queremos calcular, passo a citar as opções: unidades de comprimento, energia, temperatura e velocidade e sair. A opção sair vai fazer voltar ao menu principal. As outras opções vão entrar num outro menu onde aparecerão as opção de transformar uma unidade noutra unidade semelhante aplicando a devida fórmula, sempre que o utilizador quiser pode voltar para o menu principal sem ter que reiniciar a aplicação

Para o ponto seis a aplicação separa um nome completo já predefinido por um espaço, após isso ele escreve o ultimo nome em maiúsculos seguido pelo primeiro nome, e seguido pelas primeiras letras dos nomes seguintes em maiúsculas, tal como estão as páginas amarelas

Dll- biblioteca dinâmica.

Array- Conjunto de valores.

Todos os algoritmos criados para a solução de problemas e os menus estão introduzidos em métodos dentro de uma DLL.