



# Exercícios de Fixação - Aula 05

## Orientações gerais para resolução dos algoritmos:

- Ler o exercício cuidadosamente, até obter seu completo entendimento.
- Se for adequado, crie uma "massa" de dados para simular o ambiente de execução do programa. Não se esqueça de gerar os "valores extremos" dentro do universo dos possíveis valores (por exemplo, zero, o menor valor, o maior valor, valor negativo, etc.). Não se limite apenas à regra; analise também as exceções.
- Crie a solução mental do problema (parcial ou total se possível).
- Se o problema for complexo, trabalhe de maneira analítica, desmembrando-o em partes e resolvendo cada parte individualmente (refinamento sucessivo).
- v. Desenvolva o código correspondente ao algoritmo criado. Se você ainda tem dificuldade de entendimento, opte pelo fluxograma antes do código.
- Faça os testes de mesa com o código, simulando a resolução do problema. Se foi gerada uma massa de dados (item 2), use-a simulando seu tratamento pelo algoritmo.

## Exercícios:

50. Elaborar um programa que exiba na tela automaticamente todos os números naturais entre 0 e 100, em ordem crescente. Fazer 3 versões desta solução: uma utilizando laço com pré-teste (enquanto), outra usando laço com pós-teste (repita) e uma última usando laço finito (para).
51. Elaborar um programa que exiba na tela automaticamente todos os números naturais entre 100 e 0, em ordem decrescente. Fazer 3

versões desta solução: uma utilizando laço com pré-teste (enquanto), outra usando laço com pós-teste (repita) e uma última usando laço finito (para).

52. Elaborar um programa que exiba na tela, automaticamente, 50 valores gerados aleatoriamente, entre 10 e 99. Fazer 3 versões desta solução: uma utilizando laço com pré-teste (enquanto), outra usando laço com pós-teste (repita) e uma última usando laço finito (para).
53. Elaborar um programa que exiba na tela, automaticamente, valores gerados aleatoriamente, entre 10 e 99. Os valores deverão ser gerados indefinidamente até que surja o valor 80, que ao surgir não deve ser exibido na tela. Fazer 3 versões desta solução: uma utilizando laço com pré-teste (enquanto), outra usando laço com pós-teste (repita) e uma última usando laço finito (para).
54. Elaborar um programa que exiba, na tela, todos os caracteres visuais da tabela ASCII (do 32 ao 126) no formato "*código – caractere*". Exemplo: "80 – P". Fazer 2 versões desta solução: uma utilizando laço com pré-teste (enquanto), outra usando laço finito (para).
55. Elaborar um programa que exiba na tela automaticamente todos os múltiplos de 3 existentes entre 1 e 100. Utilizar obrigatoriamente o laço finito (para) para esta solução.
56. Elaborar um programa que exiba na tela automaticamente todos os múltiplos de um determinado número informado pelo usuário, existentes entre 1 e um limite, que também deverá ser informado pelo usuário. Fazer duas versões da solução utilizando diferentes estruturas de repetição, a sua escolha.
57. Elaborar um programa que gere, automaticamente, valores aleatórios, entre 0 e 99 e exiba na tela apenas àqueles que são múltiplos de um número informado pelo usuário. A quantidade de valores a serem gerados, também deve ser informado pelo usuário. Fazer duas versões da solução utilizando diferentes estruturas de repetição, a sua escolha.
58. Faça um programa que calcule a somatória dos dez primeiros números naturais.
59. Faça um programa que calcule a somatória dos dez primeiros números naturais pares.
60. Faça um programa que calcule a somatória dos "n" primeiros números naturais pares ou ímpares. O usuário informará através de digitação: a quantidade de números desejada ("n") e escolhe entre par o ímpar.

A partir daqui os exercícios deverão ser implementados no PascalZim.

61. Desenvolver uma máquina de soma (cálculo de somatório). O programa deverá ler uma série indefinida de valores informados pelo usuário. A leitura deverá prosseguir até que o valor zero seja informado. Ao final apresentar na tela o total do somatório.  
Implementar uma versão com cada uma das três estruturas de repetição e mais uma utilizando o desvio incondicional.
62. **Jogo da forca - parte 1:** Neste momento trataremos apenas da leitura do palpite ("chute") do usuário. Fazer um programa que realize a consistência de entrada da digitação de um palpite do usuário de modo que ele seja forçado a digitar uma letra efetivamente. Fazer 3 versões do programa, cada uma com uma das estruturas de repetição. Responder: qual achou mais adequada? Por quê?
63. [Refactoring do 24] Ler quatro valores referentes a quatro notas escolares de um aluno e imprimir uma mensagem dizendo que o aluno foi aprovado, se o valor da média escolar for maior ou igual a 7. Se o valor da média for menor que 7, solicitar a nota de exame, somar com o valor da média e obter nova média. Se a nova média for maior ou igual a 5, apresentar uma mensagem dizendo que o aluno foi aprovado em exame. Se o aluno não foi aprovado, indicar uma mensagem informando esta condição. Apresentar com as mensagens o valor da média do aluno, para qualquer condição. Atenção. Deverá ser feita a consistência dos dados de entrada para aceitar apenas notas com valores entre 0 e 10.
64. [Refactoring do 31] Elaborar um programa que efetue a leitura do nome e do sexo biológico de uma pessoa, apresentando como saída uma das seguintes mensagens: "Ilmo. Sr.", para o sexo informado como masculino ou a mensagem "Ilma. Sra. ", para o sexo informado como feminino. Apresente também abaixo da mensagem impressa o nome da pessoa. Os valores válidos para o sexo são "M" para masculino ou "F" para feminino. Não poderá ser aceito nome vazio.
65. Fazer um programa que simule um cadastro de usuário e senha para um sistema qualquer. O usuário informará um "username" e uma senha que deverá ser digitada duas vezes (confirmação de senha). Realizar as seguintes consistências:
- a. o "username" deve possuir mais do que 5 caracteres.
  - b. a senha e a confirmação da senha devem ser idênticas.
  - c. a senha deve possuir ao menos 6 caracteres e deve conter obrigatoriamente: ao menos uma letra maiúscula, um dígito numérico e um símbolo.
66. Um colega seu que trabalha no setor financeiro, mensalmente, tem

que reajustar valores de contratos recorrentes, com fatores acumulados de índices diversos, dependendo de cada contrato, tais como: IGPM, IPCA, dentre outros. Sabendo que você agora é um desenvolvedor, ele lhe solicitou que desenvolva uma calculadora para cálculo do fator de acumulado de acréscimo ou decréscimo, que deverá ser aplicado ao contrato. Ele quer digitar um a um os fatores mensais

e pressionando a tecla "ESC" ao final da série de valores, ver o resultado do fator acumulado e, também, a quantidade de fatores digitados. Quer também, ter a opção de continuar calculando novas séries de fatores sem ter que sair do programa e executá-lo novamente até que realmente queira sair. Analisando a solicitação do seu amigo você destacou alguns pontos que deverá levar em consideração na elaboração do programa:

- a. O cálculo do acumulado é feito através da multiplicação dos fatores (produtório).
- b. Só são válidos fatores positivos; menores que zero e maiores do que um.

67.[Refactoring do CS 03] Nas olimpíadas deste ano, anotou-se em cartões cada uma das medalhas de ouro, prata e bronze, obtidas por um determinado país. Faça um programa que leia esses cartões e exiba o quadro completo de medalhas obtidas pela delegação do país. O cartão com o conteúdo "F" indica que é o último da pilha.

68.[Refactoring do CS 02]: Faça um programa que leia um conjunto de notas de alunos de uma determinada turma, anotadas em cartões, calcule e exiba a média das notas dos alunos. Sabe-se que a classe possui 15 alunos.

69.Elaborar um programa que calcule e exiba as potências de 3 variando de 0 até 15, ou seja,  $3^0$ ,  $3^1$ ,  $3^2$ , ...,  $3^{14}$ ,  $3^{15}$ . O Pascal não possui uma função para cálculo de potência, mas possui uma para cálculo de exponencial. Desta forma através de propriedade matemática dos logaritmos, você pode resolver cálculo de potência e até de raízes n-ésimas da seguinte forma:  $\exp(\text{expoente} * \ln(\text{base}))$ ;

70.Altere o algoritmo do programa anterior de forma a não utilizar a função "exp".

71.Fazer um programa que dado um número inteiro positivo, informado pelo usuário, calcule n e exiba o seu fatorial.

72.Fazer um programa para calcular e exibir os "n" primeiros números da série de Fibonacci. O usuário informará o valor de n.

73.Faça um programa que leia um conjunto de notas de alunos anotadas em cartões, calcule e exiba a média das notas dos alunos, assim como a maior e a menor nota da turma. Deverá funcionar para turma com qualquer quantidade de alunos, que sempre corresponderá à quantidade de notas armazenadas nos cartões. A tecla "ESC" indica o encerramento da série de notas.

74. **Jogo Adivinha o número - parte 2:** Continue o desenvolvimento do jogo. Agora, o programa deverá continuar dando dicas a cada palpite do usuário até que ele acertar o número sorteado. Quando o usuário acertar o número gerado deverá ser apresentado o resultado do teste. Da seguinte forma:

- Em menos de 5 tentativas exibir a mensagem: "Você é muito bom, acertou em x tentativas."
- Em mais do que 5 e menos do que 9 tentativas: "Você é bom, acertou em x tentativas".
- Em mais do que 9 e menos do que 13 tentativas: "Você é mediano, acertou em x tentativas".
- Em mais do que 13: "Você é muito fraco, acertou em x tentativas"

75. Elaborar um programa que efetue o cálculo e no final apresente o somatório do número de grãos de trigo que se pode obter num tabuleiro de xadrez, obedecendo a seguinte regra: colocar um grão de trigo no primeiro quadro e, nos quadros seguintes o dobro do quadro anterior. Ou seja, no primeiro quadro coloca-se 1 grão, no segundo quadro coloca-se 2 grãos (neste momento tem-se 3 grãos), no terceiro quadro coloca-se 4 grãos (tendo nesse momento 7 grãos), até atingir o sexagésimo quarto quadro. (*exercício inspirado em uma anedota do livro "O homem que calculava" de Malba Tahan*).

76. Para cada uma das séries abaixo, elabore um programa que calcule o valor de Z. O valor de n deverá ser informado pelo usuário:

$$a) \quad z = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$

$$b) \quad z = \frac{1}{30} + \frac{2}{29} + \frac{3}{28} + \frac{4}{27} + \frac{5}{26} + \frac{6}{25} + \dots + \frac{28}{3} + \frac{29}{2} + \frac{30}{1}$$

$$c) \quad z = \frac{1}{2} - \frac{2}{4} + \frac{3}{6} - \frac{4}{8} + \frac{5}{10} - \frac{6}{12} + \dots$$

77. Um cliente utiliza uma determinada solução de GED/ECM. Nesta solução os usuários criam fichas de produtos e enviam uma série de arquivos de imagens deste dado produto. Você foi solicitado a fazer uma rotina que vá lendo uma série de dados informando o tamanho de cada arquivo em Megabytes. O programa deverá informar ao usuário um próximo arquivo estourar o limite estabelecida de 1,5 Gigabaytes, para cada ficha.

78. Fazer um programa para determinar e exibir se um número informado é primo.

79. **Jogo do "pim" infalível.** Fazer um programa que simule o jogo do "pim". Ele servirá para treinar competidores deste game. Neste jogo você vai verbalizando todos os números de 1 a 60, sendo que toda

vez que se passa por um múltiplo do número determinado como "pim", ao invés de cantar o número, devemos dizer "pim". Exemplo digamos que

o "pim" serão múltiplos de 3. Deverá ser cantado: "1, 2, pim, 4, 5, pim, ..., 58, 59, pim". Seu programa deverá simular esta situação. O usuário informará o número PIM e o programa exibirá o resultado na tela. Estabelecer um atraso a cada saída para que o jogador possa ir cantando juntamente com o programa.

80. Considere dados referentes a altura e o sexo de 40 pessoas. Fazer um algoritmo que calcule e escreva:

- a. A maior e a menor altura do grupo.
- b. A média de altura das mulheres.
- c. O número de homens.

81. Jogo do Jokenpô. Desenvolva o jogo "Papel, pedra e tesoura" entre o usuário e o computador. Após o usuário informar sua escolha, o computador gera sua opção (de maneira randômica); informa a opção gerada e compara-a com a opção do usuário, informando quem ganhou aquela jogada. Atualiza e exibe um placar que acumula os resultados de cada jogada e pergunta se o usuário deseja continuar o jogo (encerrar com ESC). Após a primeira versão criar algumas "mods" para o game.

- a. Torná-lo bilíngue (português e inglês). Esta opção deverá ser dada no início do jogo. Dica: crie constantes para os labels e mensagens.
- b. Para melhorar a aleatoriedade do computador. Ao invés de 1 a 3 gere valores entre 1 e 9 ou 1 e 12 e, reduza-os após gerados.
- c. Melhore a interface visual. Utilizando caracteres estendidos ASCII e "ASCII arts".

82. Fazer um programa que leia 10 valores e apresente-os na ordem inversa em que foram digitados.

83. Altera o algoritmo anterior, para que faça o mesmo para 10.000 valores.