



LABORATÓRIO I – AULA : Programas com Seleção – IF e CASE

Disciplina: Linguagem de Programação PASCAL
Professor responsável: Fernando Santos Osório
Semestre: 2000/2
Horário: 63

E-mail: osorio@exatas.unisinos.br
Web:
<http://www.inf.unisinos.br/~osorio/lab1.html>
Xerox : Pasta 54 – LAB. I (Xerox do C6/6)

Lista de Exercícios - Lista número: 02 - Programas com Seleção

1. Faça um programa para ler um número inteiro e exibir na tela a mensagem 'Par' se ele for um número par, ou 'Ímpar' se ele for um número ímpar. Usar a função "odd" do Pascal. Exemplo de tela de saída :

Entre com um número: 36
O número **36** é **par**.

2. Ler um número inteiro e testar se ele é positivo, negativo ou nulo, utilizando **comandos IF aninhados** (usar no máximo 2 IFs). Imprimir o resultado na tela. Exemplo de tela de saída:

Entre com 1 número: 25
O número **25** é **positivo**.

3. Ler dois números inteiros. Se o primeiro número for positivo ou igual a zero, imprimir "positivo" e somar este número ao segundo número. Se o primeiro número for negativo, imprimir "negativo" e multiplicar com o segundo. Imprimir "resultado" e o valor resultante da operação entre os dois números lidos. Exemplo de tela de saída:

Entre com 2 números: -2 5
O primeiro número é **negativo**.
O resultado da operação é: **-10**

4. Ler três números inteiros e exibir o maior deles. Se forem iguais, exibir qualquer um dos três. Exemplo de tela de saída:

Entre com 3 números: 18 23 4
O maior número é o **23**.

5. Ler três números e ordena-los de forma crescente, de maneira que a primeira variável lida contenha o menor número, a segunda o número do meio e a terceira o maior número. Exibir na tela as três variáveis com os números ordenados. Exemplo de tela de saída:

Entre com o primeiro número: 8
Entre com o segundo número: 5
Entre com o terceiro número: 6
Os números ordenados são: **5, 6 e 8**.

6. Ler três números inteiros e exibir na tela valores dispostos em ordem crescente e decrescente. Exemplo:

Entre com o 1o. número: 5
Entre com o 2o. número: -7
Entre com o 3o. número: 1

Ordem crescente : -7 1 5
Ordem decrescente: 5 1 -7

7. Escreva um programa que leia o nome e as três notas de um aluno de uma escola, na qual é utilizada uma média simples com aprovação superior a 5.0. Imprima seu boletim, como segue, indicando se o aluno foi aprovado ou reprovado. Exemplo de tela de saída:

Aluno: José da Silva
Notas: 7.6 8.2 6.9
Média: **7.56**
Resultado: **Aprovado**

8. Faça um programa para calcular a média final de um aluno da UNISINOS, numa disciplina. São lidos: número de matrícula do aluno, nota do Grau A, nota do Grau B. Caso o aluno não tenha alcançado a média 6.0, pedir ao usuário para informar a nota do Grau C, e pedir também para ele indicar qual dos graus (A ou B) que ele deseja substituir. Após a leitura dos dados do aluno, limpar a tela e exibir o número de matrícula do aluno, as suas notas e a média final deste aluno, conforme indicado no exemplo de tela abaixo:

Entre com o número de matrícula: 123456-7
Entre com a nota do Grau A: 8.2
Entre com a nota do Grau B: 4.1
Média abaixo do limite de aprovação: **5.46**
Entre com a nota do Grau C: 6.4
Qual nota (A ou B) que a prova do Grau C substitui: B

Matrícula: **123456-7**
Notas do Aluno: **8.2 4.1 6.4**
Média Final: **7.0**

9. Ler uma data de nascimento de uma pessoa fornecida através de uma *string* fornecida sob a forma DD/MM/AAAA (DD = Dia, MM = Mês, AAAA = Ano). Testar a validade desta data para saber se esta é uma data válida:

- Testar os dias válidos: dia > 0, dia <= 28 em fevereiro (29 se o ano for bissexto),
dia <= 30 em abril, junho, setembro e novembro,
dia <= 31 nos outros meses.
- Testar a validade do mês: mês > 0 e mês < 13.
- Testar a validade do ano: ano <= ano atual (constante igual a 2000).

Ao terminar de testar a validade da data, exibir na tela: "data válida" ou "data inválida", conforme o resultado dos testes citados acima.

10. Efetuar a leitura de três valores (variáveis A, B e C) e efetuar o cálculo das raízes de uma equação de segundo grau. Testar para ver se a equação possui duas raízes, uma única raiz ou se ela não possui raízes reais. Lembrete: $X = (-B \pm \sqrt{B^2 - 4AC}) / (2A)$

Exemplo de tela de saída:

Entre com o coeficiente A: 3
Entre com o coeficiente B: 6
Entre com o coeficiente C: 0
As raízes da equação são: **-2 e 0**.

11. Ler os seguintes dados de uma pessoa: nome, sexo (M ou F), idade (0 à 150) e nacionalidade (brasileira ou estrangeira). Testar a validade dos dados fornecidos, indicando se o sexo, a idade e a nacionalidade são válidos ou inválidos. Se um dos dados fornecidos for inválido indicar ao usuário. Se todos os dados forem válidos, imprimir uma mensagem como segue, onde aparecem os dados fornecidos:

“**José Silva, brasileiro** do sexo **masculino** e **maior** de idade, **está** habilitado a dirigir”, ou
“**Maria Silva, brasileira** do sexo **feminino** e **maior** de idade, **está** habilitada a dirigir”, ou
“**Luis Silva, brasileiro** do sexo **masculino** e **menor** de idade, **não está** habilitado a dirigir”.

12. Ler a data atual do computador e imprimir de forma extensa. Para ler a data utilize a função *getdate(ano, mês, dia, dia-da-semana)*, declarando as variáveis do tipo Word (inteiro positivo de 0 a 65535). Indique a utilização da biblioteca de funções “*dos*” na seção “*Uses*” do programa. A saída do programa deve ser efetuada da forma indicada logo abaixo:

Data obtida do computador: 11/08/00.

Data exibida na tela: “**Sexta-feira, dia 11 de agosto de 2000**”

13. Re-escreva o programa anterior, que exhibe a data por extenso, utilizando o comando “*case*” para testar os dias da semana e o mês.
14. Faça um programa que leia a data atual do sistema, e guarde os dados referentes a esta data em variáveis auxiliares (dia, mês, ano). Em seguida, leia uma data informada pelo usuário (dia, mês, ano), e use esta data para alterar a data do sistema operacional do computador (use o comando *setdate*). Uma vez alterada a data do sistema, leia novamente a data fornecida pelo comando *getdate*, onde estará indicado o dia da semana referente a da data fornecida pelo usuário. **Não se esqueça de restaurar a data inicial (original) do sistema antes de terminar a execução do programa!**
15. Alterar o programa anterior a fim de testar uma troca da data para 01/01/2000 e imprimir a mensagem, “Este micro está pronto para o bug do milênio” caso a data tenha sido trocada corretamente, e caso contrário imprimir “Este micro não vai funcionar corretamente no ano 2000!”. Não se esqueça de restaurar a data inicial do sistema antes de terminar a execução do programa.
16. Sugestão complementar de exercícios: Vide os exercícios 1 e 2 da Aula 03 de Programação I <http://www.inf.unisinos.br/~osorio/prog1.html>