

# Trabalho Prático 6

## Abacaxi

### Algoritmos e Estruturas de Dados I

Prazo: **25** de **outubro** de 2017

## 1 Introdução

Maria, sempre que possível, gosta de ir à feira comprar abacaxis para Joãozinho e Zezinho. Sabendo que os dois preferem os abacaxis mais doces e, querendo pagar menos, pergunta para todos os vendedores qual o preço e a doçura dos abacaxis. Como ela já é frequentadora do local, sabe exatamente qual a chance de um vendedor estar mentindo.

## 2 Tarefa

Para definir qual abacaxi possuirá maior custo benefício, Maria te contratou para, dados o preço ( $A_p$ ) e doçura ( $A_d$ ) do abacaxi e confiança ( $V_c$ ) do vendedor, definir qual é o abacaxi que ela deve comprar.

Para solucionar esse problema, você deverá definir qual o ganho real ( $A_g$ ) de cada abacaxi (dado pela Equação 1) e escolher aquele onde esse ganho é máximo.

$$A_g = \frac{A_d * V_c}{A_p} \quad (1)$$

**Você deve usar *structs* guardadas em um *array* para resolver esse problema!** Mais especificamente, você deverá usar uma `struct abacaxi` para guardar os valores de cada abacaxi, onde  $A_d$  é do tipo `int` e  $A_p$  e  $V_c$  são do tipo `float`. Caso ache necessário, você também poderá guardar o número  $A_g$ , do tipo `float`. O cálculo do abacaxi com maior ganho real deverá ser feito apenas no final da leitura de todos os abacaxis.

## 3 Entrada

A entrada é composta de várias viagens ao mercado. A primeira linha de cada viagem é um número  $N$  que define quantos abacaxis serão lidos.  $N$  nunca será maior que 1000. As próximas  $N$  linhas são compostas de um número inteiro

( $A_d$ ), representando a doçura do abacaxi. As outras  $N$  linhas são compostas de um número de ponto flutuante ( $A_p$ ), representando o preço do abacaxi (**isso para os mesmos abacaxis das  $N$  linhas anteriores!**). Por fim, as últimas  $N$  linhas são compostas de outro número de ponto flutuante ( $V_c$ ), dessa vez representando a confiança no vendedor. O final da entrada é indicado por  $N = 0$ .

## 4 Saída

Ao final de cada viagem, seu programa deverá imprimir a doçura ( $A_d$ ), o preço ( $A_p$ ) **com 2 casas decimais**, a confiança do vendedor ( $V_c$ ) **com 5 casas decimais** e o ganho real ( $A_g$ ) **com 5 casas decimais** do abacaxi escolhido. Na dúvida, siga o exemplo mostrado abaixo.

## 5 Exemplo

### Entrada

```
2
10
5
5.50
2.00
0.85
0.5
1
5
3
1.0
2
10
5
6.00
9.00
0.8
0.95
0
```

### Saída

```
10 5.50 0.85000 1.54545
5 3.00 1.00000 1.66667
10 6.00 0.80000 1.33333
```

Outros exemplos estão disponíveis na página deste trabalho no Moodle.

## 6 Avisos

Avisos mandatórios para o envio do trabalho:

- O código deve ser escrito em linguagem C
- As entradas que serão utilizadas para teste não conterão erros, então não será necessário testar a validade das mesmas
- Não utilize chamadas da função `system` (por exemplo, `system("pause")`) pois essas podem variar de acordo com o sistema operacional e os programas instalados da máquina onde o programa está rodando
- Deixe seu código bem comentado para facilitar a correção
- Não utilize espaços ou caracteres especiais nos nomes dos arquivos. Utilize apenas os caracteres de A a Z (tanto maiúsculas como minúsculas) sem acento, os números de 0 a 9 e os caracteres - (hífen), \_ (*underscore*) e . (ponto final)
- Utilize a extensão `.c` para arquivos de código e `.h` para arquivos de cabeçalho, quando aplicar
- Se for submeter os arquivos via upload, não envie um arquivo comprimido (por exemplo, `.zip`, `.rar`, etc.). Utilize os diversos campos da aba "Submissão", um para cada arquivo
- Envie apenas os arquivos `.c` e `.h`
- Não copie o trabalho de algum colega ou baixe da internet. Lembre-se que o prejudicado será você pois o aprendizado obtido nessa disciplina será utilizado durante diversas outras etapas do seu curso

## 7 Dicas

Dicas importantes para o desenvolvimento deste trabalho:

- Utilize a função `scanf` para ler as entradas do usuário
- Utilize a função `printf` para imprimir os resultados das operações
- Para utilizar *structs*, lembre-se que cada abacaxi deverá ser uma *struct* e seus dados serão guardados dentro dela
- Lembre-se que as saídas devem ser calculadas e impressas apenas ao final da viagem!
- Qualquer dúvida que tiver, utilize o fórum de dúvidas no Moodle. Inicie o assunto do tópico com a tag [TP6]
- Caso prefira, participe das monitorias toda quarta das 17h às 18h na sala 2012

## 8 Checklist

Checklist não exaustiva de passos até a entrega do trabalho:

1. Pesquise o funcionamento das funções citadas acima para facilitar o uso
2. Implemente e compile o programa
3. Teste para o exemplo dado acima. Compare as saídas para garantir o funcionamento correto
4. Faça o mesmo do item anterior para os exemplos disponibilizados na página deste trabalho no Moodle
5. Envie o trabalho pelo Moodle, onde ele será testado automaticamente para todos os casos disponíveis
6. Caso algum teste dê errado, volte ao passo 2

Bom trabalho e divirta-se!