

Primeiro Exercício Programa

MAC0115 - Física Diurno 2010

Professor Roberto M Cesar Jr
Entrega até 17/09/2010

1 Objetivo

Este programa consiste em simular a travessia de um trem entre duas estações:

1. A primeira estação está em x_1 e a segunda em x_2 .
2. O trem parte com velocidade nula e pode atingir até a velocidade máxima V_{max} .
3. A aceleração e desaceleração do trem são constantes e de módulo a .
4. O trem deve "chegar"¹ na segunda estação com velocidade zero.

2 Implementação

Os dados de entrada são x_1 , x_2 , V_{max} e a (em SI). A partir deles deve ser imprimida na tela² uma tabela mostrando o tempo decorrido, a velocidade e posição do trem; além disso deve ser exibida a diferença entre a posição final do trem e x_2 .

Para isso, o seu programa deve:

1. Fazer iterações correspondente a um intervalo de tempo δt , ou seja, calcular para o próximo tempo $t + \delta t$ a posição x e velocidade v do trem (lembre-se que não pode ultrapassar V_{max}). As fórmulas $v = v' + a.\delta t$ e $x = x' + v'.\delta t + a.\delta t^2/2$ devem ser suficientes.

¹As aspas foram usadas pois será utilizado método iterativo, que nem sempre dará o resultado esperado.

²Ou seja, na linha de comando.

2. Se chegar à velocidade V_{max} , calcular a distância de desaceleração d_{des} para que o trem comece a freiar assim que atingir ou ultrapassar³ $x_2 - d_{des}$.
3. Existirão situações onde o trem deverá desacelerar antes de atingir V_{max} . Pense neste caso...
4. Assim que a velocidade do trem voltar a 0 (zero), calcular a diferença entre a sua posição e x_2 .

Considere as estações e o trem como pontuais, ou seja, sem dimensão.

2.1 Iteração

Aqui vai uma breve explicação sobre iteração.

Nas fórmulas já apresentadas $v = v' + a.\delta t$ e $x = x' + v'.\delta t + a.\delta t^2/2$ existem:

1. x' e v' : são a posição e velocidade do trem num dado tempo t .
2. x e v : são a posição e a velocidade do trem no tempo seguinte $t + \delta t$.
3. δt : é o "incremento" no tempo do processo iterativo. Tal valor é definido pelo programa (`#define`).

Com isso, dadas as condições iniciais $x_i = x_1$ e $v_i = 0$ em $t_i = 0$, é possível calcular o movimento do trem para o tempo $t_i + \delta t$, e com esse para o tempo $t_0 + 2\delta t$, e assim por diante...

2.2 Nome de Variáveis

No seu programa **devem existir ao menos** as seguintes variáveis:

1. **x1** e **x2**: posições das estações.
2. **vmax**: velocidade maxima do trem.
3. **a**: módulo da aceleração do trem.
4. **t**: tempo decorrido.
5. **x** e **v**: posição e velocidade do trem.

³Aqui pode novamente ocorrer erro devido ao método iterativo.

6. d : distância mínima para o trem parar no instante atual t .

7. `numIt`: número da iteração atual.

Utilize `#define dT` para o incremento δt . Você pode escolher o valor que quiser de `dT`, e ver o que acontece.

3 Exemplos

Abaixo podem ser vistos dois exemplos da saída do programa:

3.1 Exmplo 1

```
x1:    10.000000m
x2:    100.000000m
xvmax: 10.000000m/s
a:      2.000000m/ss
```

Numero	Tempo (s)	Posicao (m)	Velocidade (m/s)
0	0.000000	10.000000	0.000000
1	1.000000	11.000000	2.000000
2	2.000000	14.000000	4.000000
3	3.000000	19.000000	6.000000
4	4.000000	26.000000	8.000000
5	5.000000	35.000000	10.000000
6	6.000000	45.000000	10.000000
7	7.000000	55.000000	10.000000
8	8.000000	65.000000	10.000000
9	9.000000	75.000000	10.000000
10	10.000000	84.000000	8.000000
11	11.000000	91.000000	6.000000
12	12.000000	96.000000	4.000000
13	13.000000	99.000000	2.000000
14	14.000000	100.000000	0.000000

```
Erro (x - x2): 0.000000m
```

```
Incremento de iteracao dT = 1
```

3.2 Exemplo 2

x1: 0.000000m
x2: 100.000000m
xvmax: 13.000000m/s
a: 7.000000m/ss

Numero	Tempo (s)	Posicao (m)	Velocidade (m/s)
0	0.000000	0.000000	0.000000
1	1.000000	3.500000	7.000000
2	2.000000	14.000000	13.000000
3	3.000000	27.000000	13.000000
4	4.000000	40.000000	13.000000
5	5.000000	53.000000	13.000000
6	6.000000	66.000000	13.000000
7	7.000000	79.000000	13.000000
8	8.000000	92.000000	13.000000
9	9.000000	101.500000	6.000000
10	10.000000	104.000000	0.000000

Erro (x - x2): 4.000000

Incremento de iteracao dT = 1

4 Instruções

4.1 Sobre a Elaboração

O EP pode ser elaborado por equipes de dois alunos, desde que as seguintes regras sejam respeitadas:

- Os alunos devem trabalhar sempre juntos. A ideia é que deve existir uma cooperação.
- Caso em um grupo exista um aluno com maior facilidade, este deve explicar as decisões tomadas, e o seu par deve participar e se esforçar para entender o desenvolvimento do programa. (denominamos isso de *programação em pares*, que é uma excelente prática que vocês devem se esforçar para adotar).
- Mesmo a digitação do EP deve ser feita em grupo, enquanto um digita, o outro fica acompanhando o trabalho.

- Recomendamos fortemente que o exercício seja desenvolvido da forma descrita nos itens acima, mas ele também pode ser feito individualmente.

4.2 Sobre a Avaliação

- É sua responsabilidade manter o código do seu EP em sigilo, ou seja, apenas você e seu par podem ter acesso ao código.
- No caso de EPs feitos em dupla, a mesma nota será atribuída aos dois alunos do grupo.
- **Não serão toleradas cópias!** Exercícios copiados (com eventuais disfarces) levarão à reprovação da disciplina e o encaminhamento do caso para a Comissão de Graduação.
- Exercícios atrasados não serão aceitos.
- Exercícios com erro de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
- É muito importante que seu programa tenha comentários e esteja bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa. Isto irá influenciar a sua nota.
- Caso o programa apresente resultados "estranhos" (inesperados) para eventuais dados de entrada "incorretos", haverá desconto de nota.
- As informações impressas pelo seu programa na tela devem aparecer da forma mais clara possível. Este aspecto também será avaliado.
- Uma regra básica é a seguinte: do ponto de vista do monitor responsável pela correção dos trabalhos, quanto mais convenientemente apresentado estiver o seu programa, melhor avaliado ele será.

4.3 Sobre a entrega

- O prazo de entrega é até o dia 17/09/2010.
- Caso feito em dupla: **Ambos devem submeter a versão final do código!**
- Entregar apenas um arquivo com o nome Trem.c.

- Para a entrega, utilize o Paca. Você pode entregar várias versões de um mesmo EP até o término do prazo, mas somente a última versão que permanecerá armazenada pelo sistema.
- Não serão aceitas submissões por e-mail ou atrasadas. Não deixe para a última hora, pois o sistema pode ficar congestionado, e você corre o risco de não conseguir enviar.
- Guarde uma cópia do seu EP pelo menos até o final do semestre.
- No início do arquivo, acrescente o seguinte cabeçalho:

```

/*****/
/** MAC0115 IFUSP Diurno 2010                               **/
/** Prof Roberto Cesar                                     **/
/**                                                         **/
/** Primeiro Exercicio Programa -- Trem                     **/
/**                                                         **/
/** <nome do(a) aluno(a)> <numero USP>                       **/
/** <nome do(a) aluno(a)> <numero USP>                       **/
/**                                                         **/
/** Informacoes sobre Desenvolvimento:                      **/
/** <Ambiente (Dev-C++, CodeBlocks, Gedit)>                 **/
/** <Sistema Operacional (Windows, Linux)>                  **/
/*****/

```