

Atividade Prática 2 — Algorit. e Prog. (10 pontos)

Prof. Fernando Esquírio Torres

Regras:

- A atividade ficará aberta no BlackBoard até a data limite de entrega.
 - Prazo de entrega: dia **09/11/2020** até as **11:59h (parte da manhã)**.
- A submissão da atividade deve ser feita em um **único arquivo zip**. Este arquivo deve ter o nome do aluno, ex: **John_Wick.zip**. Este zip deve conter apenas 2 arquivos-fonte de C, exatamente com os nomes ex01.c e ex02.c, cada um com o código-fonte do algoritmo correspondente.
- A atividade deve ser feita em grupos de 1, 2 ou 3 alunos. Para o trabalho feito em grupo de mais de 1 aluno, cada aluno deverá postar a sua atividade no seu portal e aos alunos deverão criar um comentário com os nomes dos alunos e RA no início do arquivo fonte com a função main.
- Códigos entregues em qualquer outra linguagem que não seja C não serão corrigidos e serão zerados. Exemplos de algumas linguagens não permitidas: Java, C++, C#, Python, JavaScript, GO, Ruby, etc.
- Caso alunos submetam códigos iguais ou muito semelhantes, será considerado plágio. Consequentemente, todos os alunos receberão zero na atividade inteira.
 - Uso um verificador de plágio, que irá rodar com todos os códigos.
 - **Cuidado ao procurar uma “solução” na internet e entregar apenas copiando, outro grupo pode ter a mesma ideia.**
- Caso seu código não compile ou apresente erros de execução (crash, loop infinito, etc.), o código será corrigido valendo 50% do valor da questão;
- O seu código será avaliado não apenas no quesito funcionamento, mas também a organização, nomenclatura de variáveis, boas práticas e qualidade dos comentários.
- Estamos fazendo uma avaliação com prazo fixo de entrega, entenda que não será possível sanar dúvidas extras por e-mail ou não haverá prorrogação de prazo. Dúvidas podem ser tiradas na aula do dia 31/10/2020 e na aula do dia 07/11/2020.

Erros comuns:

- Não use acentuação ou sinais gráficos, tanto no código quanto nos nomes dos arquivos;
- Não importe a biblioteca <locale.h> e nem a <conio.h>;
- Não retorne outro valor além de 0 na função main;
- Verifique se está inicializando corretamente todas as variáveis, principalmente os acumuladores e contadores.

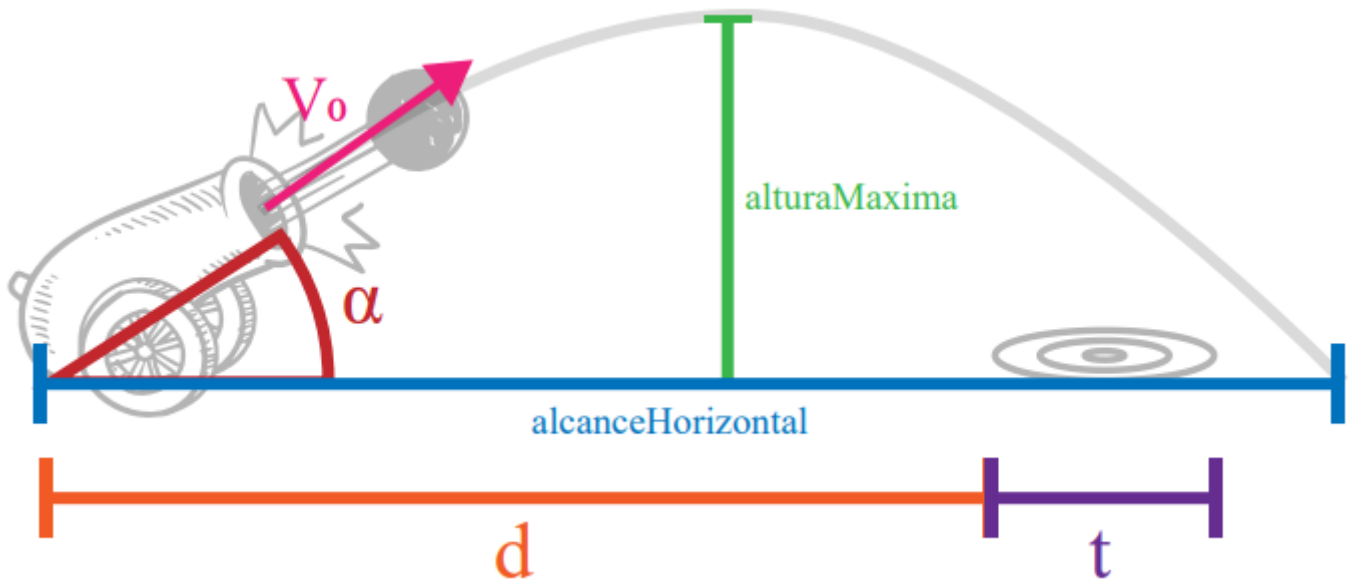
Critérios de avaliação dos upgrades nos códigos:

- 0%: Funcionalidade não é válida ou sem sentido ou funciona para situações bem específicas.
- 50%: Funcionalidade não foi completa e/ou funciona satisfatoriamente e/ou apresenta falhas em algumas situações ou funcionalidade completa sem comentários explicativos e/ou com a explicação muito simples.
- 100%: Funcionalidade funciona corretamente sem erros nos testes e com a explicação completa do que foi realizado.

Exercício 01 (5 pontos) – Empresa Sirius Cybernetics Corp.

A empresa Sirius Cybernetics Corp. quer desenvolver um programa para a simulação de lançamento de projéteis, com o objetivo de atingir um determinado alvo. Essa aplicação pode ser observada em um lançamento de uma bala de canhão, na qual para acertar o alvo é necessário definir o ângulo de inclinação do canhão (α) e a velocidade inicial (V_0) que a bala foi lançada.

Espera-se com esse programa obter o alcance horizontal, o tempo do trajeto e a altura máxima atingida por cada disparo, além de se um determinado alvo foi ou não atingido.



Funcionamento

- A tela de início do programa deve apresentar o **nome completo (Sem abreviações)** e o RA de cada integrante do grupo! (Valor: 0,5 pontos)
- Mostrar o seguinte menu para o usuário (Valor: 0,5 pontos):
 1. Simular disparo
 2. Sair
- Caso o usuário tente usar uma opção inválida, deve ser comunicado por uma mensagem, e o programa deve pedir a opção desejada novamente. A opção sair é a única forma de finalizar o programa. (Valor: 0,5 pontos)

A seguir, é detalhado o comportamento esperado de cada item do menu.

Simular disparo

Deve ser pedido ao usuário para que digite as seguintes informações abaixo sobre o disparo. **Validação:** Cada uma das entradas de dados deve ser validada. Caso o valor inserido não seja válido, deverá ser solicitado ao usuário que digite novamente aquele dado. Se o dado for válido, o programa pode prosseguir a execução, perguntando a próxima informação ao usuário.

- Velocidade Inicial (V_0), que deve ser positiva e diferente de 0 (Zero); (Valor: 0,5 pontos)
- Ângulo de inclinação do canhão (α), que será inserido pelo usuário em graus e deve estar na faixa de valores (0, 90) (intervalo aberto); (Valor: 0,5 pontos)
- Distância do alvo (d), em metros, que deverá ser maior do que zero; (Valor: 0,5 pontos)
- Tamanho do alvo (t), em metros, que deve estar na faixa de valores [20, 50] (intervalo fechado). (Valor: 0,5 pontos)

Em seguida, essa tela deve apresentar o alcance horizontal, o tempo do trajeto e a altura máxima atingida por este disparo, bem como se o disparo atingiu o alvo, ou se caiu antes ou depois dele. (Valor: 1 ponto)

Depois disso, o programa deve retornar ao menu e aguardar a escolha do usuário. (Valor: 0,5 pontos)

Informações úteis para resolver o exercício (Equações e constantes)

Considere como $g = 9,81m/s^2$ a constante de aceleração da gravidade.

Considere $\pi = 3.14$.

Para converter um valor de graus para radianos, basta multiplicá-lo por $\frac{\pi}{180}$

A altura máxima pode ser calculada utilizando a seguinte fórmula:

$$alturaMaxima = \frac{V_0^2 * \sin(\alpha)}{2g}$$

O tempo do trajeto é calculado pela fórmula:

$$tempoTrajeto = \frac{2V_0 * \sin(\alpha)}{g}$$

A fórmula do alcance horizontal é:

$$alcanceHorizontal = 2V_0^2 * \frac{\cos(\alpha) * \sin(\alpha)}{g}$$

Em C, as funções abaixo podem ser úteis (lembre-se de importar a biblioteca **math.h**):

- $\sin(x)$: seno de x, que espera x em radianos
- $\cos(x)$: cosseno de x, que espera x em radianos

Exercício 02 (5 pontos) – Empresa do Malvado Doofenshmirtz

A Empresa do Malvado Doofenshmirtz quer fazer programa para verificar se uma data é válida ou não. O programa receberá a data no formato DD/MM/AAAA, não é para fazer leituras separadas do dia, mês e ano (Valor: 1 ponto).

O programa deverá mostrar o seguinte menu para o usuário (Valor: 1 ponto):

1. Digitar Data
2. Sair

O programa terá que fazer as seguintes verificações:

1. Ano digitado está entre [1900, 2020] (intervalo fechado) e se o ano é ou não bissexto. (Valor: 1 ponto)
2. Mês digitado está entre [1, 12] (intervalo fechado). (Valor: 1 ponto)
3. Dia digitado está no intervalo correto dependendo do mês (Valor: 1 ponto):
 - a. Entre [1, 31] (intervalo fechado) para os meses de janeiro, março, maio, julho, agosto, outubro e dezembro.
 - b. Entre [1, 30] (intervalo fechado) para os meses de abril, junho, setembro e novembro.
 - c. Entre [1, 28] (intervalo fechado) para o mês de fevereiro e ano não bissexto.
 - d. Entre [1, 29] (intervalo fechado) para o mês de fevereiro e ano bissexto.