

## Instituto de Computação - Unicamp

### MC102 - Algoritmos e Programação de Computadores

# Laboratório 06 - Soma Acumulada

---

Prazo de entrega: **07/04/2017 23:59:59**

Peso: **1**

*Professor:* Eduardo C. Xavier

*Professor:* Guido Araújo

*Monitor:* Arthur Pratti Dadalto

*Monitor:* Cristina Cavalcante

*Monitor:* Klairton de Lima Brito

*Monitor:* Luís Felipe Mattos

*Monitor:* Paulo Finardi

*Monitor:* Paulo Lucas Rodrigues Lacerda

*Monitor:* Pedro Alves

*Monitor:* Renan Vilas Novas

*Monitor:* Vinicius de Novaes Guimarães Pereira

## Descrição

---

Você é um engenheiro da Frota Estelar, servindo na espaçonave USS Enterprise juntamente com o Dr. Spock e o Capitão Christopher Pike. Você descobriu que existe um vazamento de antimatéria no condutor de combustível da Enterprise, mas antes que pudesse localizar a fonte do vazamento, uma frota de naves inimigas Klingon apareceu e começou a atacar.

Os escudos da espaçonave foram ativados, mas não irão aguentar muito

tempo. O plano é juntar combustível suficiente para fazer a espaçonave entrar em dobra espacial e fugir. O problema é que com o vazamento de antimatéria, parte do combustível está sumindo, pois cada unidade de antimatéria aniquila uma unidade de combustível.

Dado que o escudo da espaçonave tem um tempo de duração  $t$ , que para entrar em dobra é necessário uma quantidade de combustível  $c$ , e que para cada unidade de antimatéria, uma unidade de combustível é cancelada, você deve determinar se é possível juntar combustível suficiente para fugir antes que o escudo pare de funcionar.

## Objetivo

---

O objetivo deste laboratório é criar um programa que, dado o tempo máximo  $t$ , em segundos, que o escudo irá funcionar, a quantidade  $c$  de combustível necessária para fazer a Enterprise entrar em dobra espacial, e uma sequência de números, um número por segundo, indicando a quantidade de combustível passando pelo condutor (valores positivos) ou antimatéria (valores negativos), deve-se determinar se é possível juntar combustível suficiente para fugir antes que o escudo pare de funcionar.

Note que para cada unidade de antimatéria, é aniquilada uma unidade de combustível. Considere por exemplo a sequência de números abaixo indicando a quantidade de combustível e antimatéria passando pelo condutor de combustível:

-2 -4 3 3 3 4 -1 -1 4

No final do tempo 3 o total de combustível é 0, pois houve um fluxo de -6 de antimatéria e 6 de combustível, já ao final do tempo 5, o total de combustível é 7, e ao final do tempo 8 o total de combustível é 9.

## Entrada

---

A entrada consiste de três linhas, na primeira linha será dado um número

inteiro positivo indicando o tempo `t` em segundos, restante até que o escudo da espaçonave falhe, na segunda linha será dado um outro inteiro positivo indicando a quantidade de combustível `c` necessária para fazer a espaçonave entrar em dobra, e na última linha, uma sequência de números inteiros, positivos e negativos, contendo o fluxo de combustível (valores positivos) ou antimatéria (valores negativos) a cada segundo.

*Dica para a linguagem C:* leia a entrada da seguinte forma:

```
scanf("%d %d", &t, &c);
```

para o tempo e a quantidade de energia, e:

```
scanf("%d", &f);
```

para cada valor da sequência de números indicando o fluxo de combustível ou antimatéria no condutor.

**Importante:** Você pode assumir que há números suficientes na sequência de números da terceira linha para que ou atinja-se a quantidade de combustível necessário ou o escudo pare de funcionar e a nave seja destruída.

## Saída

---

A saída deverá conter duas linhas: a primeira linha deverá conter a palavra `sim` se for possível juntar combustível suficiente para fugir antes que o escudo acabe, e a palavra `nao` caso contrário; a segunda linha deve conter o instante de tempo em que foi possível fugir ou então o instante de tempo que a nave foi destruída.

*Dica para a linguagem C:* imprima a saída da seguinte forma:

```
printf("sim\n%d\n", i);
```

ou

```
printf("nao\n%d\n", t+1);
```

**Importante:** O tempo inicia em zero segundos. Note que ao final de um determinado tempo  $t$ , se for juntado combustível suficiente, a nave consegue fugir apenas no instante de tempo seguinte  $t+1$ . Da mesma forma, se o escudo para de funcionar no tempo  $t$ , a espaçonave será destruída no segundo seguinte  $t+1$ .

## Exemplos

---

### Teste 01

#### Entrada

```
5
10
7 -5 3 6 -1
```

#### Saída

```
sim
4
```

### Teste 02

#### Entrada

```
6
20
6 8 -10 5 9 -4
```

### Saída

```
nao
7
```

#### Teste 03

### Entrada

```
5
15
10 2 -6 4 5
```

### Saída

```
sim
5
```

Para mais exemplos, consulte os [testes abertos no Susy](#).

## Observações

---

- O número máximo de submissões é **10**;
- O seu programa deve estar completamente contido em um único arquivo denominado `lab06.c` ;
- Para a realização dos testes do SuSy, a compilação dos programas desenvolvidos em C irá considerar o comando:  

```
gcc -std=c99 -pedantic -Wall -o lab06 lab06.c ;
```
- Você deve incluir, no início do seu programa, uma breve descrição dos objetivos do programa, da entrada e da saída, além do seu nome e do seu RA;
- Indente corretamente o seu código e inclua comentários no decorrer

do seu programa.

## Critérios importantes

---

Independentemente dos resultados dos testes do SuSy, o não cumprimento dos critérios abaixo implicará em nota zero nesta tarefa de laboratório.

- Para os programas em linguagem C, o único header aceito para essa tarefa é o `stdio.h`.