

ALGPROG – Primeira Lista de Exercícios do Laboratório

Regras

- A data limite de entrega do trabalho é domingo, 31 de Maio de 2015 até as 23:59:59.
 - O trabalho deve ser realizado em dupla ou trios (não serão aceitos trabalhos individuais, nem em grupos com mais de três alunos).
 - Os programas-fonte (apenas os fontes – executáveis são rejeitados por programas de anti-vírus para email) devem ser entregues por correio eletrônico para seu professor com cópia para si mesmo.
 - Trabalhos iguais de grupos diferentes receberão nota zero.
 - A lista é composta de 5 exercícios.
 - No campo do assunto deve-se colocar a turma a que o grupo pertence (por exemplo: 2BEPN).
 - NO CORPO DO EMAIL, DEVE CONSTAR OS NOMES E R.A.S DOS MEMBROS DO GRUPO.
 - No programa, deve constar os nomes e R.A.s dos membros do grupo como comentário nas primeiras linhas.
 - Cada um dos programas que for aceito no SPOJ poderá ser premiado com pontuação extra, a critério do professor.
-

Sistema de avaliação de alunos de um curso.

Os alunos são identificados em números sequenciais de 10000 a 10020.

O sistema armazena duas notas: P1 e P2 com valores válidos entre 0.0 e 10.0.

A média considerada deve ser: $0,4P1 + 0,6P2$ e deve totalizar 6,0 para aprovação.

O sistema deverá ter as opções:

- 1 - Receber notas P1 de todos os alunos.
- 2 - Receber notas P2 de todos os alunos.
- 3 - Receber nota P1 de um aluno.
- 4- Receber nota P2 de um aluno.
- 5 - Verificar a condição de um aluno.
- 6 - Listar alunos em intervalo.
- 7 - Listar todos os alunos.
- 0 – Termina o programa.

As ações são as seguintes:

Opção 1 – Recebe sequencialmente cada nota. O programa deve apresentar a identificação do aluno para o qual está recebendo a nota, e deve ter a opção de finalizar a entrada a qualquer momento.

Opção 2 – Recebe sequencialmente cada nota. O programa deve apresentar a identificação do aluno para o qual está recebendo a nota, e deve ter a opção de finalizar a entrada a qualquer momento .

Opção 3 – Recebe a nota de um aluno, para isso o sistema solicita o número do aluno e verifica se o aluno existe.

Opção 4 – Recebe a nota de um aluno, para isso o sistema solicita o número do aluno e verifica se o aluno existe.

Opção 5 – Solicita o número do aluno, apresenta todas as suas notas, a média e diz se o aluno foi aprovado ou reprovado.

Opção 6 – O sistema solicita um intervalo de números e apresenta todos os alunos, suas médias e situação.

Opção 7 – Faz o mesmo para todos os alunos, COM PAGINAÇÃO.

Opção 0 – Termina o programa.

TODOS os elementos devem ter entrada consistida/validada, isto é:

- as notas devem ser fornecidas em intervalos válidos
- a lista dos alunos devem ser fornecidas com intervalos válidos
- os erros devem ser informados ao usuário

Valores aproximados para "e", "π" e $\sqrt{2}$.

- a. A aproximação da função e^x em série de Taylor é dada pela expressão:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots, \quad -\infty < x < \infty$$

O valor de "e" pode ser obtido calculando a série para o valor de $x = 1.0$

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}$$

- b. A aproximação da função $\arctan(x)$ em série de Taylor é dada por:

$$\tan^{-1}(x) = \sum_{k=0}^{\infty} \frac{(-1)^k x^{1+2k}}{1+2k} \quad \text{for } |x| < 1$$

O valor de "π" pode ser obtido calculando a série para o valor de $x = 1.0$

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \dots + \frac{(-1)^n}{2n+1}$$

- c. O valor de \sqrt{A} pode ser obtido através da fórmula:

$$x_{n+1} = \frac{A - x_n^2}{2 \cdot x_n}$$

O valor de $\sqrt{2}$ pode ser obtido calculando a série para $A = 2$ e o valor de $x_0 = 1.0$.

Escreva um programa em linguagem C que determine o valor mínimo de "n" para os valores de "e", "π" e $\sqrt{2}$ utilizando as fórmulas acima que apresente um erro inferior a 10^{-5} . Utilize, como referência, os valores dados por M_E, M_PI e M_SQRT2 da biblioteca math.h.

SPOJ Problem Set (obi)

812. Bits Trocados

Problema: BIT

As Ilhas Weblands formam um reino independente nos mares do Pacífico. Como é um reino recente, a sociedade é muito influenciada pela informática. A moeda oficial é o Bit; existem notas de B\$ 50,00, B\$10,00, B\$5,00 e B\$1,00. Você foi contratado(a) para ajudar na programação dos caixas automáticos de um grande banco das Ilhas Weblands.

Tarefa

Os caixas eletrônicos das Ilhas Weblands operam com todos os tipos de notas disponíveis, mantendo um estoque de cédulas para cada valor (B\$ 50,00, B\$10,00, B\$5,00 e B\$1,00). Os clientes do banco utilizam os caixas eletrônicos para efetuar retiradas de um certo número inteiro de Bits.

Sua tarefa é escrever um programa que, dado o valor de Bits desejado pelo cliente, determine o número de cada uma das notas necessário para totalizar esse valor, de modo a minimizar a quantidade de cédulas entregues. Por exemplo, se o cliente deseja retirar B\$50,00, basta entregar uma única nota de cinquenta Bits. Se o cliente deseja retirar B\$72,00, é necessário entregar uma nota de B\$50,00, duas de B\$10,00 e duas de B\$1,00.

Entrada

A entrada é composta de vários conjuntos de teste. Cada conjunto de teste é composto por uma única linha, que contém um número inteiro positivo v , que indica o valor solicitado pelo cliente. O final da entrada é indicado por $v = 0$.

Saída

Para cada conjunto de teste da entrada seu programa deve produzir três linhas na saída. A primeira linha deve conter um identificador do conjunto de teste, no formato "Teste n ", onde n é numerado a partir de 1. Na segunda linha devem aparecer quatro inteiros I , J , K e L que representam o resultado encontrado pelo seu programa: I indica o número de cédulas de B\$50,00, J indica o número de cédulas de B\$10,00, K indica o número de cédulas de B\$5,00 e L indica o número de cédulas de B\$1,00. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo

Entrada:

```
1
72
0
```

Saída:

```
Teste 1
0 0 0 1

Teste 2
1 2 0 2
```

Restrições

$0 \leq V \leq 10000$ ($V = 0$ apenas para indicar o fim da entrada)

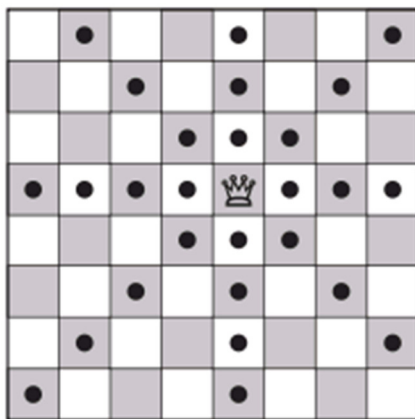
Adicionado por:	Wanderley Guimarães
Data:	2006-04-19
Tempo limite:	1s
Tamanho do fonte:	50000B
Memory limit:	256MB
Cluster:	Pyramid (Intel Pentium III 733 MHz)
Linguagem permitida:	Todas exceto: AWK CLOJ ERL F# GO JS NODEJS PERL 6 PYTH 3.2.3 n
Origem:	Olimpiada Brasileira de Informatica 2000

SPOJ Problem Set (regionais)

3240. Dama

Problema: DAMA

O jogo de xadrez possui várias peças com movimentos curiosos: uma delas é a dama, que pode se mover qualquer quantidade de casas na mesma linha, na mesma coluna, ou em uma das duas diagonais, conforme exemplifica a figura abaixo:



O grande mestre de xadrez *Kary Gasparov* inventou um novo tipo de problema de xadrez: dada a posição de uma dama em um tabuleiro de xadrez vazio (ou seja, um tabuleiro 8×8 , com 64 casas), de quantos movimentos, no mínimo, ela precisa para chegar em outra casa do tabuleiro?

Kary achou a solução para alguns desses problemas, mas teve dificuldade com outros, e por isso pediu que você escrevesse um programa que resolve esse tipo de problema.

Entrada

A entrada contém vários casos de teste. A primeira e única linha de cada caso de teste contém quatro inteiros $X1$, $Y1$, $X2$ e $Y2$ ($1 \leq X1, Y1, X2, Y2 \leq 8$). A dama começa na casa de coordenadas $(X1, Y1)$, e a casa de destino é a casa de coordenadas $(X2, Y2)$. No tabuleiro, as colunas são numeradas da esquerda para a direita de 1 a 8 e as linhas de cima para baixo também de 1 a 8. As coordenadas de uma casa na linha X e coluna Y são (X, Y) .

O final da entrada é indicado por uma linha contendo quatro zeros.

Saída

Para cada caso de teste da entrada seu programa deve imprimir uma única linha na saída, contendo um número inteiro, indicando o menor número de movimentos necessários para a dama chegar em sua casa de destino.

Exemplo

Entrada

```
4 4 6 2
3 5 3 5
5 5 4 3
0 0 0 0
```

Saída

```
1
0
2
```

Adicionado por: [Wanderley Guimarães](#)
Data: 2008-10-25
Tempo limite: 1s
Tamanho do fonte: 50000B
Memory limit: 256MB
Cluster: [Pyramid \(Intel Pentium III 733 MHz\)](#)
Linguagem permitida: Todas exceto: AWK CLOJ ERL F# GO JS NODEJS PERL 6 PYTH 3.2.3 n SCALA SED TCL
Origem: Primeira fase da Maratona de Programação - 2008

<http://br.spoj.com/problems/PRIMO/>

SPOJ Problem Set (obi)

3828. Primo

Problema: PRIMO

Tarefa

Dado um inteiro N , verifique se N é primo.

Entrada

A entrada é composta por um único caso de teste, composto por uma única linha que contém o inteiro N .

Saída

Seu programa deve produzir uma única linha, contendo a palavra "sim", se N for primo, e "nao", caso contrário (note a ausência de acentuação).

Restrições

$$|N| < 2^{31}$$

Exemplo

Entrada

7

Saída

sim

Entrada

10

Saída

nao

Adicionado por:	Wanderley Guimarães
Data:	2009-02-07
Tempo limite:	1s
Tamanho do fonte:	50000B
Memory limit:	256MB
Cluster:	Pyramid (Intel Pentium III 733 MHz)
Linguagem permitida:	Todas exceto: AWK CLOJ ERL F# GO JS NODEJS PERL 6 PYTH 3.2.3 n
Origem:	SCALA SED TCL
	Treino para OBI de 2006 - Fábio Moreira & Daniel Fleischman