

Cursos de Verão - Instituto de Matemática e Estatística - USP

Introdução à Programação

Prova 2 - Entrega até 28/01/2013 às 08:00 pelo PACA

Professor: Neuton de Oliveira Braga Junior

Instruções:

1. NÃO ATRASE A ENTREGA. ATRASOS NÃO SERÃO TOLERADOS.
2. Leitura e Interpretação é fundamental na resolução dos problemas. Leia com atenção todas as questões e responda elas de acordo com o que é pedido.
3. Se um problema estiver muito complicado de fazer, tente primeiro escrever no papel o algoritmo com a lógica de programação que você identificou e somente depois tente escrever o código C.
4. Resolva as questões utilizando boas práticas de programação em C (identação, nomenclaturas, etc).
5. Nas questões que envolvem elaboração de programas, utilize a linguagem **C** no padrão **ansi** e somente os recursos da linguagem vistos em sala. Além disso, o código de tais questões devem ser entregues em arquivos *.c com o código da questão no nome, por exemplo, a resolução da questão P01 deve ser entregue com o arquivo P01.c e assim por diante (Padrão de arquivo mostrado abaixo).
6. Nas questões subjetivas ou de simulação, entreguem um arquivo pdf, doc, docx etc com a resolução de todas.
7. Essa é como uma prova em sala e é PROIBIDO comunicar com colegas sobre a mesma, mas é permitido consultar material colocado no Paca ou anotações. Não seguir qualquer uma das instruções poderá implicar penalizações da nota.

```
1  /*
2  *  Introducao a Programacao – VERA0 2013 – IME – USP
3  *  Prof. Neuton de Oliveira Braga Jr
4  *  2 Prova
5  *
6  *  Resolucao da Questao <Coloque o codigo da questao, ex, Pxx>
7  *  Aluno: <Complete com seu nome aqui>
8  *  Turma: <Diga sua turma, se eh Manha ou Noite>
9  */
10
11 #include <stdio.h>
12
13 /* Inclua a biblioteca math.h, se for usa-la */
14
15 /* Declare funcoes se for necessario */
16
17 int main() {
18     printf("Esse eh somente um codigo modelo de entrega da 2 prova.\n");
19
20     return 0;
21 }
```

BOA SORTE!

P00 (valor 0.0) - Questão L00 da Lista:

Caso alguma de suas soluções esteja correta e seja uma das melhores resoluções entregues dessa lista, você gostaria que fosse divulgada para que outros colegas possam aprender com você? Sim ou Não?

P01 (valor 1.0) - Questão L73b da Lista:

Implemente um programa onde algumas das execuções ocorrem conforme mostrado abaixo:

<pre>\$./L73b Digite um numero: 3 --+ +-- +--+ +--+ +--+ +--+ __ </pre>	<pre>\$./L73b Digite um numero: 4 --+ +-- +--+ +--+ +--+ +--+ __ </pre>
--	--

P02 (valor 1.0) - Questão L75 da Lista:

Escreva um programa que recebe vários pontos (x, y) do plano cartesiano e depois mostra quantos pontos tem cada quadrante dentre os mostrados.

Exemplo de execução:

```
$ ./L75
Digite o ponto 1 (ou 0 0 para terminar): 1.5 2.5
Digite o ponto 2 (ou 0 0 para terminar): 0 5.5
Digite o ponto 3 (ou 0 0 para terminar): -5.5 -2.4e5
Digite o ponto 4 (ou 0 0 para terminar): -1.1 -2.5
Digite o ponto 5 (ou 0 0 para terminar): 0 0

Foram contabilizados 4 pontos onde:
1 ponto é do quadrante 1
2 pontos são do quadrante 4
1 ponto não é de nenhum quadrante
```

Considere o código em ANSI C a seguir para responder as questões P03 e P04.

```

1  /*
2  *  Arquivo L81-82.c
3  */
4  #include <stdio.h>
5
6  int main() {
7      int n, termo, i, res = 0;
8
9      printf("Quantos numeros tem a sequencia? ");
10     scanf("%d",&n);
11     printf("Digite a sequencia: ");
12
13     for (i = 0; i < n; i++) {
14         scanf("%d", &termo);
15         if (i % 2 == 0) {
16             res += termo;
17         } else {
18             res -= termo;
19         }
20     }
21
22     printf("O resultado do calculo eh %03d\n", res);
23
24     return 0;
25 }

```

P03 (valor 1.0) - Questão L81 da Lista:

Simule a execução do programa desse código construindo uma tabela com os valores das variáveis *i*, *termo* e *res* e, descubra qual valor é retornado (no lugar de ?) em cada uma das execuções abaixo:

a)

```

$ ./L81-82
Quantos numeros tem a sequencia? 5
Digite a sequencia: 32 11 15 14 55
O resultado do calculo eh ?

```

b)

```

$ ./L81-82
Quantos numeros tem a sequencia? 3
Digite a sequencia: 852 192 33
O resultado do calculo eh ?

```

c)

```

$ ./L81-82
Quantos numeros tem a sequencia? 4
Digite a sequencia:
18 35
24 78
O resultado do calculo eh ?

```

d)

```

$ ./L81-82
Quantos numeros tem a sequencia? 2
Digite a sequencia: 15
17
32 21
O resultado do calculo eh ?

```

P04 (valor 1.0) - Questão L82 da Lista:

Escreva esse código, compile em sua máquina e faça testes verificando:

a) Para que serve o 03 entre o % e o d no comando de formato para a variável *res* no printf? Tente colocar outros números para ver o que acontece (como, por exemplo, 04, só 4 etc).

b) Todas as simulações feitas na questão L81 estavam certas? O que a simulação da L81-d) tem de diferente das outras?

c) Altere o código para que ao invés do usuário digitar a quantidade de números da sequência, ele já comece digitando a sequência e quando ele digitar zero indicará que a sequência acabou.

P05 (valor 2.0) - Questão L87 da Lista:

Escreva um programa que leia um número natural maior que zero e menor que 3999, e mostre sua representação em algarismos romanos considerando que as principais regras para escritas de números romanos são¹:

1. São sete as letras básicas com os seguintes valores: I = 1; V = 5; X = 10; L = 50; C = 100; D = 500 e M = 1.000;
2. Se uma letra se repete, repete-se o valor: XX = 20; CCC = 300;
3. As letras I, X, C e M podem ser repetidas no máximo três vezes em seguida; as letras V, L e D não se repetem;
4. Se um valor maior precede outro menor, ambos se somam: LX = 60; CV = 105;
5. Se um valor menor precede outro maior, ele é deduzido do segundo: XC = 90 (100 - 10); CD = 400 (500 - 100).

Exemplos de alguns números: I = 1, II = 2, III = 3, IV = 4, V = 5, VI = 6, VII = 7, VIII = 8, IX = 9, X = 10, XI = 11, XIV = 14, XIX = 19, XX = 20, XXIX = 29, XXX = 30, XXXIX = 39, XL = 40, L = 50, LIX = 59, LX = 60, LXX = 70, LXXX = 80, XC = 90, XCIX = 99, C = 100, CC = 200, CCXCVII = 297, CCCXXVI = 326, CDXCIX = 499, D = 500, DCVIII = 608, DCCXIX = 719, DCCCLXXIV = 874, CMI = 901, CMXCIX = 999, M = 1.000, MCDLXXXVI = 1.486, MD = 1.500, MCMLXXXVIII = 1.988.

P06 (valor 1.0) - Questão L94 da Lista:

Muitas vezes, quando estamos querendo resolver algum problema, nos deparamos com sequências numéricas que são conjuntos (ou grupos) de números que estão escrito em alguma ordem geralmente obedecendo alguma lógica.

Uma vez conhecendo a lógica da sequência numérica, é possível encontrar o termo que aparece em qualquer posição.

Diante disso, para cada uma das sequências numéricas a seguir, descubra a lógica delas e implemente um programa onde mostra o termo na posição n informada pelo usuário:

- a) 7, 10, 8, 11, 9, 12, ...
- b)² 1, 1, 1, 2, 4, 7, 13, 24, 44, 81, 149, ...

P07 (valor 1.0) - Questão L97 da Lista:

Sabendo que, na medida de ângulos, 180 graus correspondem a π radianos. Faça um programa que converta de radianos para graus (e vice-versa) utilizando o valor do π fornecido no rodapé da página da lista anterior a esta.

¹Regras obtidas em <http://www.estadao.com.br/manualredacao/esclareca/numerosromanos.shtml>

²Essa sequência é conhecida como Sequência de Tribonacci.

Análise o código a seguir para responder a questão 08.

```

1  /*
2  *   Arquivo L98-99.c
3  *
4  *   Comando de compilacao:
5  *
6  *   gcc -Wall -pedantic -ansi L98-99.c -o L98-99 -lm
7  */
8  #include <stdio.h>
9  #include <math.h>
10
11 int main() {
12     double m, res = 0;
13     int n, i;
14
15     printf(" Digite a quantidade de termos: ");
16     scanf("%d", &n);
17     printf(" Digite o expoente: ");
18     scanf("%lf", &m);
19
20     for (i = 1; i <= n; i++) res += (pow(i,m));
21     printf(" O resultado da soma eh: %f\n", res);
22
23     return 0;
24 }

```

P08 (valor 1.0) - Questão L98 da Lista:

Análise o código acima e diga qual seu objetivo.

P09 (valor 1.0) - Questão L100 da Lista:

Escreva um programa que recebe vários pontos (x, y) do plano cartesiano e, depois mostra o ponto mais distante da origem e a distância dele para ela, e o ponto mais próximo da origem e a distância dele para ela.

P10 (valor 1.0) - Questão L103 da Lista:

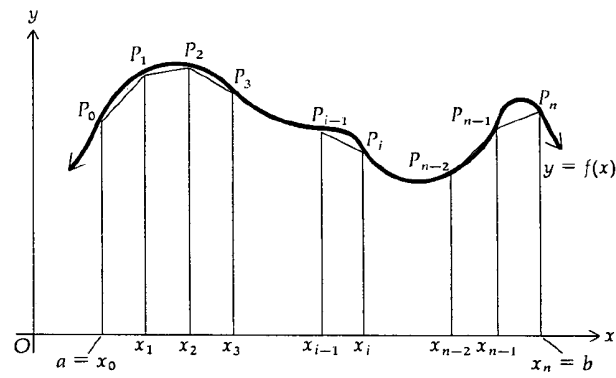
Escreva um programa onde o usuário informa as dimensões da base maior, da base menor e a altura do trapézio e é mostrado a área desse trapézio.

P11 (valor 2.0) - Questão L104 da Lista:

Há várias aplicações onde é necessário calcular a área no gráfico de uma função $f(x)$ (em relação ao eixo das abcissas) em um intervalo fechado $[a, b]$ específico.

Uma possível forma de obter valor aproximado para área de uma função $f(x)$ qualquer no intervalo $[a, b]$ é dividir tal intervalo em n partes onde cada uma corresponde a um trapézio, calcular as áreas de cada trapézio e encontrar a soma total dessas “areazinhas” que corresponderá ao valor aproximado encontrado³. Veja a ilustração abaixo.

³Rascunho da Regra dos Trapézios utilizada em integração numérica para cálculo aproximado de integrais definidas num intervalo. Leia algo em http://en.wikipedia.org/wiki/Trapezoidal_rule



Baseando-se nessa ideia exposta, escreva um programa permita o usuário informar o intervalo para o qual seja realizado o cálculo da área e também permita o usuário informar em quantas partes o intervalo deve ser dividido para a função:

$$f(x) = \frac{\lfloor \ln x \rfloor}{x}$$