

Universidade Federal de Goiás  
Instituto de Informática  
Introdução à Programação – 2020-2  
Prova P3

Prof. Celso Camilo  
Prof. Vagner Sacramento

### 1 - Maior par e menor ímpar – 3,0 pontos

Dada uma sequência de N números entre 0 e 100, determine e imprima o valor do maior número par e o valor do menor número ímpar encontrado.

#### Entrada

O programa contém vários casos de teste. A primeira linha da entrada contém o número de casos de teste. Para cada caso de teste há duas linhas. A primeira linha de cada caso de teste possui um inteiro N;  $1 \leq N \leq 10000$ , representando a quantidade de números da sequência. Na segunda linha de cada caso de teste haverá N números inteiros variando de 0 a 100, separados entre si por um espaço.

#### Saída

Para cada caso de teste o programa deve gerar duas linhas. A primeira linha contém o valor do maior número par encontrado na sequência, se existir. Se não existir um número par na sequência, o programa deve imprimir a frase: "Nao ha valores pares.". A segunda linha contém o valor do menor número ímpar encontrado na sequência, se existir algum ímpar. Se não existir um número ímpar, o programa deve imprimir a frase: "Nao ha valores impares."

#### Entrada

```
5
20
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
4
1 3 5 7
6
2 30 90 40 30 4
10
1 4 8 9 7 5 3 2 1 8
7
1 3 4 8 10 2 12
```

---

**Saída**

20

1

Nao ha valores pares.

1

90

Nao ha valores impares.

8

1

12

1

## 2 - Inversão de Letras - 3,0 pontos

Uma professora alfabetizou seus alunos, mas quer treiná-los mais com a manipulação de letras nas palavras. Ela pensou em uma brincadeira que geralmente as crianças gostam: inverter as palavras. Ela pretende fornecer para as crianças uma lista de frases com palavras escritas de forma invertida. Cada criança deve rescrever cada frase com as letras das palavras na ordem correta. Ela quer que você escreva um programa que leia frases escritas em ordem invertida e que as escreva na ordem correta. Ela pretende usar seu programa para gerar um gabarito de correção da lista de exercícios que ela quer criar.

### Entrada

A entrada é formada por zero ou mais linhas. Uma linha da entrada contém uma frase escrita com as letras invertidas. A entrada termina por fim de arquivo. Você pode assumir que toda string lida terá no máximo 499 caracteres.

### Saída

Para cada linha lida na entrada, deve ser escrita a frase correspondente com as letras na ordem correta.

### Exemplo

#### Entrada

sahnilab e sotilurip maroda sacnairc  
saditrewni oas saditrevid sartel moc sesarf

---

#### Saída

criancas adoram pirulitos e balinhas  
frases com letras divertidas sao invertidas

---

### 3 - Strcmp() Nerd - 4,0 pontos

Nerdildo está aprendendo a programar usando strings na linguagem C. Ele quer criar a sua própria versão da função strcmp, a qual ele vai denominar strcmpnerd(). Ele quer que sua função tenha a mesma funcionalidade da função strcmp() encontrada em string.h. Apenas o nome da função deve ser diferente. Nerdildo então procurou na internet a especificação da função strcmp() para implementar a função strcmpnerd() com as mesmas características. Ele encontrou as seguintes informações:

**Protótipo:** `int strcmp( char *s1, char* s2);`

A função inicia comparando os primeiros caracteres das duas strings. Se eles forem iguais entre si, a função compara o par seguinte de caracteres (um de cada string) até encontrar um par cujos caracteres diferem entre si ou encontrar um caractere nulo ('\0'). O valor retornado pela função depende do motivo do término da comparação e pode ser:

- -1, se os caracteres na posição i nas duas strings diferem entre si e s1[i] possui código na tabela ASCII inferior ao código de s2[i];
- 0, se as duas strings apontadas por s1 e s2 forem iguais;
- 1, se se os caracteres na posição i nas duas strings diferem entre si e s1[i] possui código na tabela ASCII superior ao código de s2[i];

O problema é que Nerdildo não consegue programar essa função e pede a você que escreva essa função para ele. Além de implementar a função strcmpnerd() seu programa deve ler vários casos de teste para mostrar que a função está implementada corretamente, de acordo com o especificado acima para a função strcmp().

#### Entrada

A primeira linha da entrada contém o número n,  $n > 0$  de casos de teste. Cada caso de teste contém duas linhas. A primeira linha corresponde à primeira string a ser comparada e a segunda linha corresponde à segunda string. Você pode assumir que toda string lida terá no máximo 499 caracteres.

#### Saída

Para cada caso de teste seu programa deve imprimir em uma linha um dos três possíveis valores: 0, -1, ou 1, correspondendo ao retorno da função strcmpnerd() ao comparar a primeira linha com a segunda do caso de teste correspondente na entrada.

#### IMPORTANTE

Essa questão além de ser avaliada pelo Sharif, será também avaliada pelo professor. A nota definitiva da questão será a dada pelo professor. As soluções que não implementarem a função, mas fizerem uso da função strcmp() terão nota ZERO, mesmo que o Sharif aceite a solução como correta.

**Entrada**

5

um caso de teste

um Caso de teste

outro caso de teste

outro caso de testf

mais um caso de teste

mais um caso de teste

Antonio Pereira

Antonia Pereira

Antonia Ferreira

Antonio

---

**Saída**

1

-1

0

1

-1

---