

Laboratório MIPS - 5

Rank Sort

*Imagine que **VOCÊ é o ChatGPT** e que algum estudante de Computação lhe peça um algoritmo simples de ordenação chamado Rank Sort. Sua resposta poderia ser algo como:*

O algoritmo *rank sort* (ou *counting sort*) implementa um tipo de ordenação onde não há comparação entre os elementos e o resultado é colocado num segundo vetor. Considerando um vetor desordenado **a**, a ideia do *rank sort* é contar o número de valores menores que **a[i]**. Essa contagem é o *rank* ou "classificação" desse valor e corresponderá ao índice final no vetor ordenado. Assim, o valor já classificado será colocado em outro vetor no local **b[rank] = a[i]**.

Algoritmo em C:

```
for (i = 0; i < n; i++) { // para cada número no vetor desordenado
    x = 0;
    for (j = 0; j < n; j++) // conta quantos menores
        if (a[i] > a[j]) x++;
    b[x] = a[i]; // copia o número para a posição correspondente
}
```

O estudante então lhe pede uma versão em MIPS com um vetor de 15 posições. Você é o ChatGPT, deve conseguir implementar este simples algoritmo, não? Implemente este código em MIPS e ajude o aluno!

Entrada

A entrada consiste em 15 valores positivos e não repetidos a serem ordenados.

Saída

Imprima os valores ordenados.

Exemplos

Exemplo de entrada 1

5
3
7
1
9
8
10
2
0
4
6
12
11
14
13

Exemplo de saída 1

SPIM Version 8.0 of January 8, 2010
Copyright 1990-2010, James R. Larus.
All Rights Reserved.
See the file README for a full copyright notice.
Loaded: /usr/lib/spim/exceptions.s
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14