

Estrutura de Dados Avançada

Daniel de Sousa Moraes
danielmoraes14@gmail.com

Count Sort

- Quando se tem um intervalo conhecido e pequeno. Ex: $[0,10]$, $[0,100]$
- Conta as ocorrências de um determinado valor ordenando-os a partir dessas informações
- Apenas números inteiros

Count Sort

```
countSort(V, C)
  for i=0, 100 do
    C[i] = 0
  end
  for i=0,#v do
    C[ v[i] ] = C[ v[i] ]+1
  end
  j = 1
  for i=0,100 do
    while C[i]>0 do
      v[j] = i
      j = j+1
      C[i] = C[i]-1
    end
  end
end
```

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

C=

0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

i

C=

0	0	0	1	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

i

C=

0	0	0	1	0	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---

0 1 2 3 4 5 6 7 8 9

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

i

C=

0	0	0	1	0	1	0	1	0	0
0	1	2	3	4	5	6	7	8	9

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

i

C=

0	0	0	1	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

i

C=

0	1	0	1	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

i

C=

0	1	0	2	0	1	0	1	1	0
---	---	---	---	---	---	---	---	---	---

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

j

C=

0	1	0	2	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

3	5	7	8	1	3
---	---	---	---	---	---

j

C=

0	1	0	2	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	5	7	8	1	3
---	---	---	---	---	---

j

C=

0	0	0	2	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	5	7	8	1	3
---	---	---	---	---	---

j

C=

0	0	0	2	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	7	8	1	3
---	---	---	---	---	---

j

C=

0	0	0	1	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	8	1	3
---	---	---	---	---	---

j

C=

0	0	0	0	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	8	1	3
---	---	---	---	---	---

j

C=

0	0	0	0	0	1	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	5	1	3
---	---	---	---	---	---

j

C=

0	0	0	0	0	0	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	5	1	3
---	---	---	---	---	---

j

C=

0	0	0	0	0	0	0	1	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	5	7	3
---	---	---	---	---	---

j

C=

0	0	0	0	0	0	0	0	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	5	7	3
---	---	---	---	---	---

j

C=

0	0	0	0	0	0	0	0	1	0
0	1	2	3	4	5	6	7	8	9

i

Count Sort

V=

1	3	3	5	7	8
---	---	---	---	---	---

j

C=

0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---

0 1 2 3 4 5 6 7 8 9

i

Count Sort

V=

1	3	3	5	7	8
---	---	---	---	---	---

j

C=

0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---

0 1 2 3 4 5 6 7 8 9

i

Bibliografia

Cormen, Thomas H. et al. Algoritmos.; [tradução Arlete Simille]. 3ª ed
- Rio de Janeiro - Elsevier, 2011.

Carlos de Salles Soares Neto – Notas de Aula da Disciplina de
Algoritmos I - UFMA