

Enumeration Sort

Peterson Wagner Kava de Carvalho

Enumeration Sort

Ordenação por ranks

input

10	5	20	2	8
----	---	----	---	---

rank

3	1	4	0	2
---	---	---	---	---

result

2	5	8	10	20
---	---	---	----	----



Enumeration Sort

10	5	20	2	8
----	---	----	---	---

10	+1	+1	+1	
5	+1			
20	+1	+1	+1	+1
2				
8	+1	+1		



Código sequencial

```
void enumeration_sequencial (long unsigned int *array, long unsigned int *rank, int N)
{
    for (int i = 0; i < N-1; ++i)
        for (int j = i+1; j < N; ++j)
            if (array[i] >= array[j])
                rank[i]++;
            else
                rank[j]++;
}
```

Código paralelo

```
for (int i = my_rank * (n/procs_num); i < (my_rank+1) * (n/procs_num); ++i)
{
    for (int j = i+1; j < n; ++j)
        if (array[i] >= array[j])
            rank[i]++;
        else
            rank[j]++;
}

printf("process %d/%d done\n", my_rank+1, procs_num);

if (my_rank != 0)
    MPI_Send(rank, n, MPI_INT, 0, STD_TAG, MPI_COMM_WORLD);
else
{
    int *rank_received = calloc (n*procs_num, sizeof(int));

    for (int p = 1; p < procs_num; p++)
    {
        memset (rank_received, 0, n*sizeof(int));
        MPI_Recv(rank_received, n, MPI_INT, MPI_ANY_SOURCE, MPI_ANY_TAG, MPI_COMM_WORLD, &status);

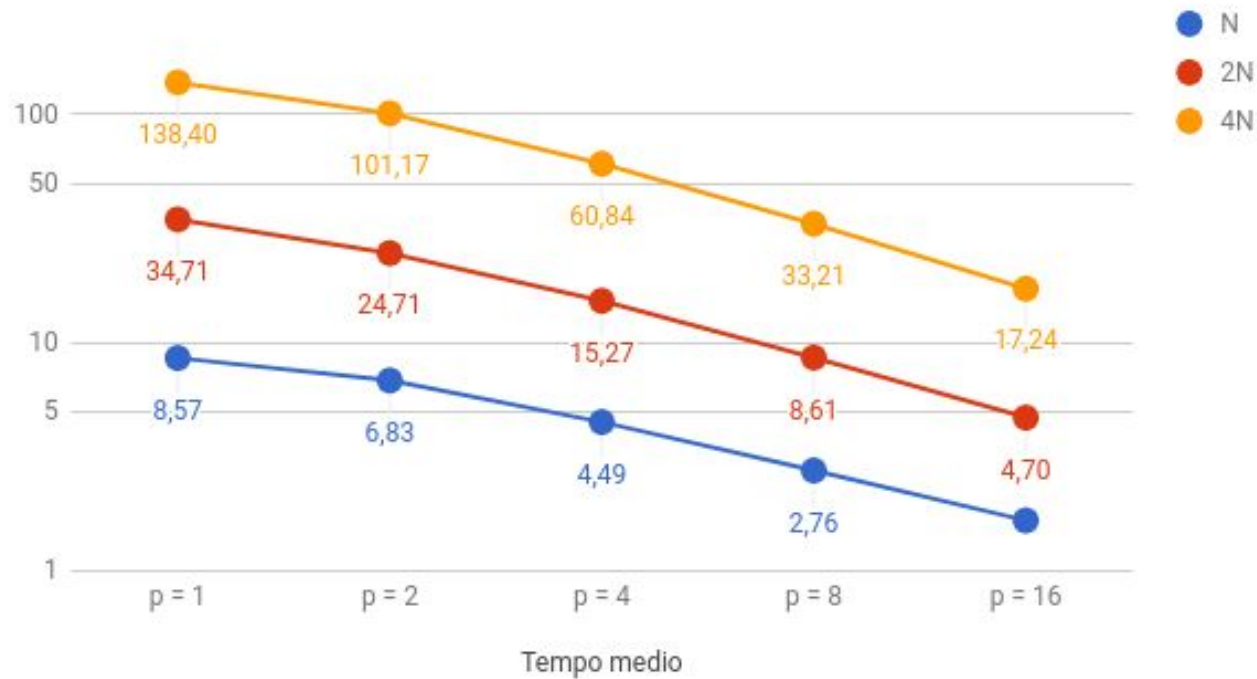
        for (int i = 0; i < n; ++i)
            rank[i] += rank_received[i];
    }

    int *result = calloc (n*procs_num, sizeof(int));

    for (int i = 0; i < n; ++i)
        result[rank[i]] = array[i];

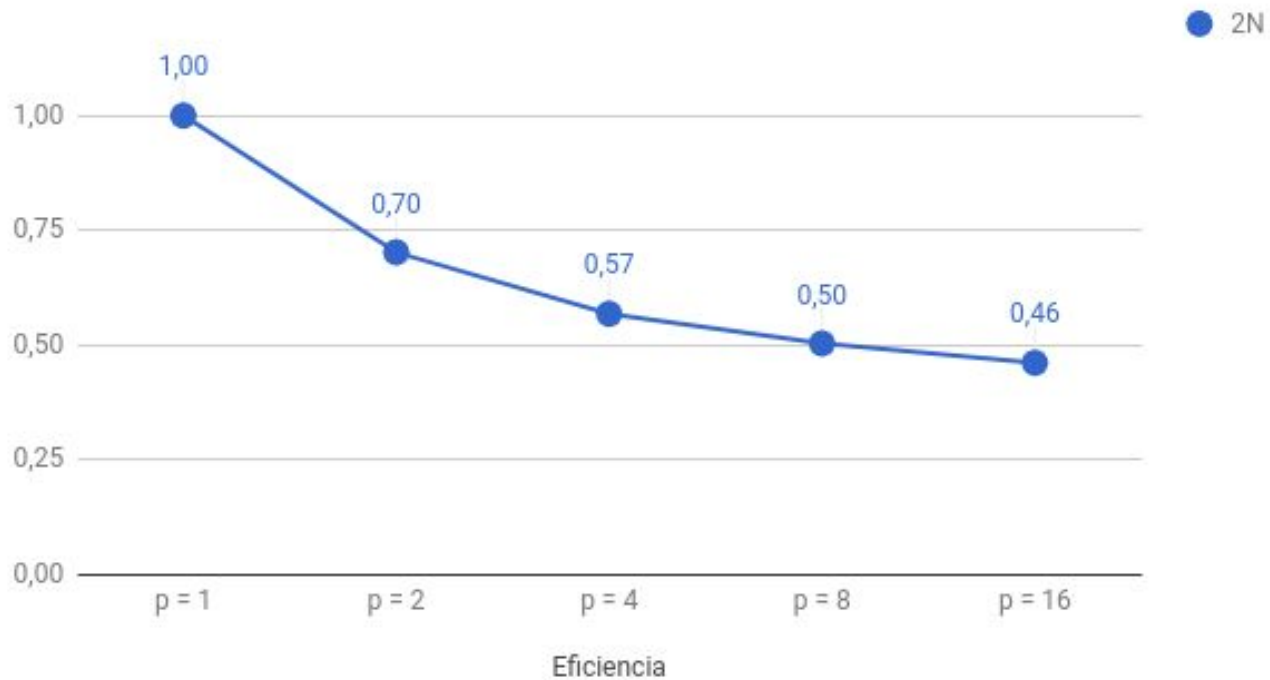
    FILE *result_fp = fopen("saida.txt", "w+");
    for (int i = 0; i < n; ++i)
        fprintf(result_fp, "%d ", result[i]);
}
```

Tempo medio



Desvio Padrao	p = 1	p = 2	p = 4	p = 8	p = 16
N	0,11	0,22	0,18	0,13	0,10
2N	1,71	1,06	0,96	0,36	0,31
4N	6,06	3,12	1,48	0,74	0,42

Eficiência (teste de escalabilidade forte)



Teste de escalabilidade fraca

Eficiência	p = 1	p = 2	p = 4	p = 8	p = 16
N	1,000	0,627	0,477	0,389	0,322
2N	1,000	0,702	0,568	0,504	0,462
4N	1,000	0,684	0,569	0,521	0,502
8N				0,493	
16N					0,445

Estimativa de bytes transferidos

Por iteração:
nenhuma

Execução total:
 $N^2 * \text{sizeof}(\text{int})$



Estimativa de instruções executadas

Por iteração:
 $(N-1)$

Execução total:
 $(N/p) * (N-1)$



Desempenho

Fontes de ganho:

pouquíssima comunicação (apenas no final para união dos resultados)

Fontes de queda:

divisão de trabalho não-uniforme

