HỌC VIỆN KỸ THUẬT QUÂN SỰ

KHOA CÔNG NGHỆ THÔNG TIN

****

**BÀI TẬP MÔN HỌC**

PHÂN TÍCH VÀ THIẾT KẾ GIẢI THUẬT

**Đề 4:**

***Thuật toán Bucket Sort***

*Giáo viên hướng dẫn:* ***Hà Đại Dương***

*Học viên thực hiện:* ***Trần Tuấn Phong***

*Lớp:* ***ATBMTTK50***

Hà Nội, 01/2017

HỌC VIỆN KỸ THUẬT QUÂN SỰ

KHOA CÔNG NGHỆ THÔNG TIN

****

**BÀI TẬP MÔN HỌC**

PHÂN TÍCH VÀ THIẾT KẾ GIẢI THUẬT

**Đề 4:**

***Thuật toán Bucket Sort***

*Giáo viên hướng dẫn:* ***Hà Đại Dương***

*Học viên thực hiện:* ***Trần Tuấn Phong***

*Lớp:* ***ATBMTTK50***

Hà Nội, 01/2017

**MỤC LỤC**

[**THUẬT TOÁN BUCKET SORT** 3](#_Toc502598363)

[**1.** **Bài toán** 3](#_Toc502598364)

[**2.** **Ý tưởng** 3](#_Toc502598365)

[**3.** **Mô tả chi tiết thuật toán** 3](#_Toc502598366)

[**4.** **Đánh giá độ phức tạp của thuật toán** 3](#_Toc502598367)

[**5.** **Thực hiện các bước của thuật toán** 4](#_Toc502598368)

[***Bộ dữ liệu 1:*** 4](#_Toc502598369)

[***Bộ dữ liệu 2:*** 5](#_Toc502598370)

**THUẬT TOÁN BUCKET SORT**

1. **Bài toán**

Cho mảng gồm n phần tử A[1..n], sắp xếp mảng A theo thứ tự tăng dần.

1. **Ý tưởng**

Bucket Sort hay còn gọi là Bin Sort là thuật toán sắp xếp, hoạt động bằng cách phân bố các phần tử của 1 dãy input vào 1 số các bucket (cái xô). Mỗi bucket được sắp xếp riêng, sử dụng các thuật toán khác hoặc có thể áp dụng đệ quy thuật toán Bucket Sort cho mỗi bucket. Nó là 1 sắp xếp phân bố, có họ hàng với Radix Sort. Bucket Sort có thể thực hiện bằng những so sánh và vì thế nên cũng có thể xem xét bằng 1 thuật toán sắp xếp so sánh. Độ phức tạp tính toán liên quan đến số bucket.

1. **Mô tả chi tiết thuật toán**

Mô phỏng lại qui trình trên:

Cho:

* Input A: **a[0]**, **a[2]**, …, **a[n-1]**
* Range A: các phần tử A nằm trong khoảng [0,range-1]
* Số bucket: bucket đánh số từ 0 đến (số bucket - 1)

Thuật toán sẽ thực hiện theo các bước:

* Tạo ra các bucket rỗng với số lượng bằng số bucket cho trước.
* Đưa các phần tử của A vào các bucket.
* Sắp xếp trong từng bucket.
* Lấy các phần tử trong các bucket theo thứ tự.

1. **Đánh giá độ phức tạp của thuật toán**

n là số phần tử, k là range A

* Nếu range A < số bucket: mỗi bucket có nhiều nhất 1 phần tử, do đó không phải sắp xếp trong mỗi bucket

Độ phức tạp O(n+k)

* Nếu range A >= số bucket: mỗi bucket có thể có nhiều phần tử, cần sắp xếp trong mỗi bucket. Trường hợp xấu nhất, tất cả phần tử nằm trong cùng 1 bucket. Độ phức tạp bằng độ phức tạp sắp xếp n phần tử.

Độ phức tạp O(nlogn)

1. **Thực hiện các bước của thuật toán**

***Bộ dữ liệu 1:***

Cho dãy số a: { 1; 10; 6; 4; 2}

Dãy có 5 số

Range a bằng 11.

Số bucket bằng 3

* Độ dài mỗi bucket = 10/3 = 3.
* Phân các phần tử vào các bucket

|  |  |  |
| --- | --- | --- |
| Bucket | Khoảng bucket |  |
| 0 | [0,2] | 1 2 |
| 1 | [3,5] | 4 |
| 2 | [6,10] | 10 6 |

* Sắp xếp theo các lô

|  |  |  |
| --- | --- | --- |
| Bucket | Khoảng bucket |  |
| 0 | [0,2] | 1 2 |
| 1 | [3,5] | 4 |
| 2 | [6,10] | 6 10 |

* Lấy ra các phần tử theo thứ tự bucket và thứ tự đã xếp trong bucket

A = {}

Bucket[0]: A = {1;2}

Bucket[1]: A = {1;2;4}

Bucket[2]: A = {1;2;4;6;10}

***Bộ dữ liệu 2:***

Cho dãy số a: { 1001; 725; 1245; 5; 32523; 324; 99998; 15421; 60000}

Dãy có 9 số

Range a bằng 99999.

Số bucket bằng 12

* Độ dài mỗi bucket = 99999/12 = 8333.
* Phân các phần tử vào các bucket

|  |  |  |
| --- | --- | --- |
| Bucket | Khoảng bucket |  |
| 0 | [0,8332] | 1001 725 1245 5 324 |
| 1 | [8333,16665] | 15421 |
| 2 | [16666,24998] |  |
| 3 | [24999,33331] |  |
| 4 | [33332,41664] | 35523 |
| 5 | [41665,49997] |  |
| 6 | [49998,58330] |  |
| 7 | [58330,66663] | 60000 |
| 8 | [66664,74996] |  |
| 9 | [74997,83329] |  |
| 10 | [83330,91662] |  |
| 11 | [91662,99998] | 99998 |

* Sắp xếp theo các lô

|  |  |  |
| --- | --- | --- |
| Bucket | Khoảng bucket |  |
| 0 | [0,8332] | 5 324 725 1001 1245 |
| 1 | [8333,16665] | 15421 |
| 2 | [16666,24998] |  |
| 3 | [24999,33331] |  |
| 4 | [33332,41664] | 35523 |
| 5 | [41665,49997] |  |
| 6 | [49998,58330] |  |
| 7 | [58330,66663] | 60000 |
| 8 | [66664,74996] |  |
| 9 | [74997,83329] |  |
| 10 | [83330,91662] |  |
| 11 | [91662,99998] | 99998 |

* Lấy ra các phần tử theo thứ tự bucket và thứ tự đã xếp trong bucket

A = {}

Bucket[0]: A = {5;324;725;1001;1245}

Bucket[1]: A = {5;324;725;1001;1245;15421}

Bucket[2]: A = {5;324;725;1001;1245;15421}

Bucket[3]: A = {5;324;725;1001;1245;15421}

Bucket[4]: A = {5;324;725;1001;1245;15421;35523}

Bucket[5]: A = {5;324;725;1001;1245;15421;35523}

Bucket[6]: A = {5;324;725;1001;1245;15421;35523}

Bucket[7]: A = {5;324;725;1001;1245;15421;35523;60000}

Bucket[8]: A = {5;324;725;1001;1245;15421;35523;60000}

Bucket[9]: A = {5;324;725;1001;1245;15421;35523;60000}

Bucket[10]: A = {5;324;725;1001;1245;15421;35523;60000}

Bucket[11]: A = {5;324;725;1001;1245;15421;35523;60000;99998}