

[Trang Chủ \(https://laptrinh tudau.com/\)](https://laptrinh tudau.com/)[Giới Thiệu](#)[Liên Hệ](#)[Khóa Học^{-40%} \(https://laptrinh tudau.com/khoa-hoc/\)](https://laptrinh tudau.com/khoa-hoc/)[← Quảng cáo của Google](#)[Chặn quảng cáo này](#)[Tại sao có quảng cáo này? ⓘ](#)

Bài Viết

[Tìm kiếm phần tử trong danh sách liên kết đơn C/C++ \(https://laptrinh tudau.com/tim-kiem-phan-tu-trong-danh-sach-lien-ket-don-c-c/\)](https://laptrinh tudau.com/tim-kiem-phan-tu-trong-danh-sach-lien-ket-don-c-c/)[Xóa phần tử trong danh sách liên kết đơn C/C++ \(https://laptrinh tudau.com/xoa-phan-tu-danh-sach-lien-ket-don/\)](https://laptrinh tudau.com/xoa-phan-tu-danh-sach-lien-ket-don/)[Đảo ngược danh sách liên kết đơn C/C++ \(https://laptrinh tudau.com/dao-nguoc-danh-sach-lien-ket-don-c-c/\)](https://laptrinh tudau.com/dao-nguoc-danh-sach-lien-ket-don-c-c/)[Bài tập danh sách liên kết đơn bằng C/C++ \(https://laptrinh tudau.com/bai-tap-danh-sach-lien-ket-don-bang-c-c/\)](https://laptrinh tudau.com/bai-tap-danh-sach-lien-ket-don-bang-c-c/)

Danh Sách Liên Kết Đôi

[Cài đặt danh sách liên kết đôi C/C++ – Doubly Linked List \(https://laptrinh tudau.com/danh-sach-lien-ket-doi/\)](https://laptrinh tudau.com/danh-sach-lien-ket-doi/)[Thêm phần tử vào danh sách liên kết đôi C/C++ \(https://laptrinh tudau.com/them-phan-tu-cc%89-vao-danh-sach-lien-ket-doi-c-c/\)](https://laptrinh tudau.com/them-phan-tu-cc%89-vao-danh-sach-lien-ket-doi-c-c/)[Nhập xuất trong danh sách liên kết đôi C/C++ \(https://laptrinh tudau.com/nha-cc%a3p-xuat-trong-danh-sach-lien-ket-doi-c-c/\)](https://laptrinh tudau.com/nha-cc%a3p-xuat-trong-danh-sach-lien-ket-doi-c-c/)[Tìm kiếm phần tử trong danh sách liên kết đôi C/C++ \(https://laptrinh tudau.com/tim-kiem-phan-tu-trong-danh-sach-lien-ket-doi-c-c/\)](https://laptrinh tudau.com/tim-kiem-phan-tu-trong-danh-sach-lien-ket-doi-c-c/)[Xóa phần tử trong danh sách liên kết đôi C/C++ \(https://laptrinh tudau.com/xoa-phan-tu-trong-danh-sach-lien-ket-doi-c-c/\)](https://laptrinh tudau.com/xoa-phan-tu-trong-danh-sach-lien-ket-doi-c-c/)[Bài tập danh sách liên kết đôi sử dụng C/C++ \(https://laptrinh tudau.com/bai-tap-danh-sach-lien-ket-doi-su-dung-c-c/\)](https://laptrinh tudau.com/bai-tap-danh-sach-lien-ket-doi-su-dung-c-c/)

Cấu Trúc Ngăn Xếp Stack

[Cấu trúc dữ liệu ngăn xếp stack trong C/C++ \(https://laptrinh tudau.com/ngan-xep-stack/\)](https://laptrinh tudau.com/ngan-xep-stack/)[Cài đặt ngăn xếp stack bằng mảng C/C++ \(https://laptrinh tudau.com/cai-dat-ngan-xep-stack-bang-mang-c-c/\)](https://laptrinh tudau.com/cai-dat-ngan-xep-stack-bang-mang-c-c/)[Cài đặt ngăn xếp stack bằng danh sách liên kết C/C++ \(https://laptrinh tudau.com/cai-dat-ngan-xep-stack-bang-danh-sach-lien-ket-c-c/\)](https://laptrinh tudau.com/cai-dat-ngan-xep-stack-bang-danh-sach-lien-ket-c-c/)[Bài tập stack cài đặt bằng mảng trong C/C++ \(https://laptrinh tudau.com/bai-tap-stack-cai-dat-bang-mang-trong-c-c/\)](https://laptrinh tudau.com/bai-tap-stack-cai-dat-bang-mang-trong-c-c/)[Bài tập stack cài đặt bằng danh sách liên kết C/C++ \(https://laptrinh tudau.com/bai-tap-stack-cai-dat-bang-danh-sach-lien-ket-c-c/\)](https://laptrinh tudau.com/bai-tap-stack-cai-dat-bang-danh-sach-lien-ket-c-c/)

Cấu Trúc Hàng Dội Queue

[Cấu trúc dữ liệu hàng đợi queue trong C/C++ \(https://laptrinh tudau.com/hang-doi-queue/\)](https://laptrinh tudau.com/hang-doi-queue/)[Cài đặt hàng đợi queue bằng mảng C/C++ \(https://laptrinh tudau.com/cai-dat-hang-doi-queue-bang-mang-c-c/\)](https://laptrinh tudau.com/cai-dat-hang-doi-queue-bang-mang-c-c/)[Cài đặt hàng đợi queue bằng danh sách liên kết C/C++ \(https://laptrinh tudau.com/cai-dat-hang-doi-queue-bang-danh-sach-lien-ket-c-c/\)](https://laptrinh tudau.com/cai-dat-hang-doi-queue-bang-danh-sach-lien-ket-c-c/)[Bài tập hàng đợi queue cài đặt bằng mảng trong C/C++ \(https://laptrinh tudau.com/bai-tap-hang-doi-queue-cai-dat-bang-mang-trong-c-...](https://laptrinh tudau.com/bai-tap-hang-doi-queue-cai-dat-bang-mang-trong-c-...)[Bài tập hàng đợi cài đặt bằng danh sách liên kết trong C/C++ \(https://laptrinh tudau.com/bai-tap-hang-doi-cai-dat-bang-danh-sach-lien-k...](https://laptrinh tudau.com/bai-tap-hang-doi-cai-dat-bang-danh-sach-lien-k...)

Cây Nhị Phân Tìm Kiếm

[Cấu trúc dữ liệu cây trong C/C++ – Tree data structure \(https://laptrinh tudau.com/cau-truc-du-lieu-cay/\)](https://laptrinh tudau.com/cau-truc-du-lieu-cay/)

Cài đặt cấu trúc cây nhị phân tìm kiếm trong C/C++ (<https://laptrinh tudau.com/cai-dat-cau-truc-cay-nhi-phan-tim-kiem-trong-c-c/>)

Thêm phần tử vào cây nhị phân trong C/C++ (<https://laptrinh tudau.com/them-phan-tu-vao-cay-nhi-phan-trong-c-c/>)

Duyệt cây nhị phân tìm kiếm trong C/C++ (<https://laptrinh tudau.com/duyet-cay-nhi-phan-tim-kiem-trong-c-c/>)

Tìm kiếm phần tử cây nhị phân tìm kiếm trong C/C++ (<https://laptrinh tudau.com/tim-kiem-phan-tu-cay-nhi-phan-tim-kiem-trong-c-c/>)

Xóa phần tử cây nhị phân tìm kiếm trong C/C++ (<https://laptrinh tudau.com/xoa-phan-tu-cay-nhi-phan-tim-kiem-trong-c-c/>)

Thuật toán tìm kiếm

Tìm kiếm tuyến tính (Linear Search) trong C/C++ (<https://laptrinh tudau.com/tim-kiem-tuyen-tinh-linear-search-trong-c/>)

Tìm kiếm nhị phân (Binary Search) trong C/C++ (<https://laptrinh tudau.com/tim-kiem-nhi-phan-binary-search-trong-c/>)



<

Quảng cáo của Google

Chặn quảng cáo này

Tại sao có quảng cáo này? ⓘ

	▲					
			139,000 đ	214,000 đ		
	▼					
		281,600 đ	819,000 đ	130,000 đ		

Trang chủ » (<https://laptrinhtrivau.com>) CTD> » (<https://laptrinhtrivau.com/cau-truc-du-lieu-giai-thuat/>) Xóa phần tử trong danh sách liên kết đơn C/C++

Xóa phần tử trong danh sách liên kết đơn C/C++

Tag: danh sách liên kết đơn (<https://laptrinhtrivau.com/tag/danh-sach-lien-ket-don/>) delete list (<https://laptrinhtrivau.com/tag/delete-list/>) Single linked list (<https://laptrinhtrivau.com/tag/single-linked-list/>)

Thẻ nhớ microSD Sandisk Ult...

Ưu điểm nổi bật của thẻ nhớ MicroSDXC SanDisk 667x SDSQUNR-064G-GN3MN...

QC Tiki

Trong danh sách liên kết đơn, để xóa một phần tử (hay một Node) trong danh sách sẽ có 4 trường hợp xảy ra ra:

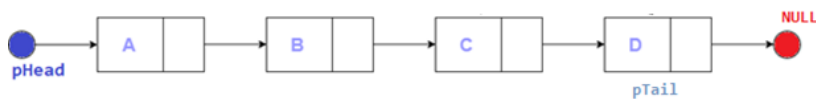
- Xóa phần tử ở đầu danh sách liên kết đơn
- Xóa phần tử ở cuối danh sách liên kết đơn
- Xóa phần tử có khóa k trong danh sách liên kết đơn
- Hủy toàn bộ danh sách liên kết đơn

Nội Dung [ẩn]

- 1.Xóa phần tử ở đầu danh sách liên kết đơn
- 2.Xóa phần tử ở cuối danh sách liên kết đơn
- 3.Xóa phần tử có khóa k trong danh sách liên kết đơn
- 4.Hủy toàn bộ danh sách liên kết đơn

1.Xóa phần tử ở đầu danh sách liên kết đơn

Danh sách đơn ban đầu của tôi bao gồm các phần tử (hay các node) bên dưới:

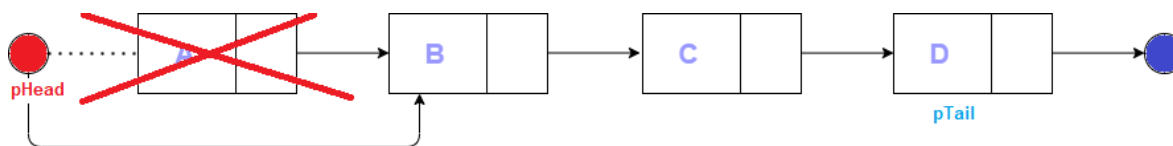


		281,600 đ	192,900 đ	
	139,000 đ	103,350 đ	313,000 đ	

Để thực hiện xóa phần tử đầu tiên của danh sách trên ta chỉ cần thực hiện 3 bước:

- **Bước thứ nhất:** Tạo một **node p** và gán bằng **pHead** hiện tại của danh sách: **p = ds.pHead**
- **Bước thứ hai:** Đặt lại phần tử pHead của danh sách bằng phần tử kế sau pHead hiện tại của danh sách: **ds.pHead = ds.pHead->next**
- **Bước thứ ba:** Gán **node p** được gán bằng **pHead** ở bước thứ nhất trở về NULL: **p->next = NULL**
- **Bước thứ tư:** Xóa đi node p được gán bằng **pHead** ở bước thứ nhất: **delete p**

Để hiểu rõ hơn các bước trên, hãy xem hình dưới đây:



	139,000 ₺	144,000 ₺	130,000 ₺	281,600 ₺
--	-----------	-----------	-----------	-----------

Hàm **void XoaDau**(*LIST &ds*) dưới đây thực hiện xóa phần tử (hay node) đầu tiên trong danh sách:

```
1. void XoaDau(LIST &ds){
2.     //tao node p
```

```

3.     NODE *p = new NODE;
4.     //gan p bang node pHead dau tien cua danh sach
5.     p = ds.pHead;
6.     //thay doi lai pHead cua danh sach
7.     ds.pHead = ds.pHead->next;
8.     //gan node p ban dau tro den NULL
9.     p->next = NULL;
10.    //xoa node p
11.    delete p;
12. }

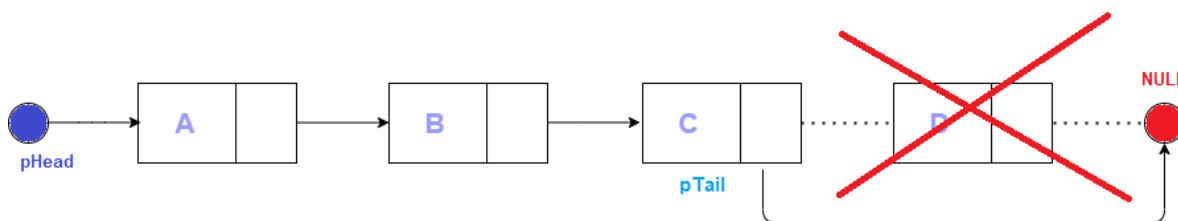
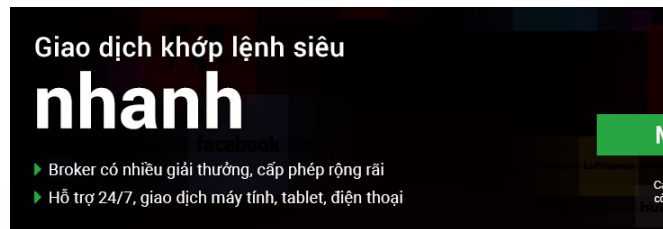
```

2.Xóa phần tử ở cuối danh sách liên kết đơn

Để xóa phần tử ở cuối danh sách liên kết đơn, ta cần thực hiện 4 bước sau:

- **Bước đầu tiên:** Tạo một **node k** và thực hiện duyệt toàn bộ phần tử có trong danh sách liên kết đơn
- **Bước thứ hai:** Kiểm tra duyệt, nếu thực hiện duyệt đến phần tử cuối của danh sách nghĩa là **k == ds.pTail** thì thực hiện xóa phần tử đó: **delete ds.pTail;**
- **Bước thứ ba:** Trỏ phần tử đứng trước **pTail** bằng NULL: **k->next = NULL;**
- **Bước thứ tư:** Thay đổi lại **pTail** của danh sách bằng **node k**

Xem hình dưới đây để hiểu rõ hơn các bước trên:



Hàm **void XoaCuoi(LIST &ds)** dưới đây thực hiện xóa phần tử (hay node) đầu tiên trong danh sách:

```

1. void XoaCuoi(LIST &ds)
2. {
3.     //duyet cac phan tu co trong danh sach
4.     for(NODE *k = ds.pHead; k != NULL; k = k ->next)
5.     {
6.         //neu duyet den phan tu pTail cuoi trong danh sach
7.         if(k->next == ds.pTail)
8.         {
9.             //xoa phan tu cuoi
10.            delete ds.pTail;
11.            //tro phan tu truoc phan tu cuoi ve NULL
12.            k->next = NULL;
13.            //thay doi lai phan tu cuoi pTail cua danh sach
14.            ds.pTail = k;
15.        }
16.    }
17. }

```

3.Xóa phần tử có khóa k trong danh sách liên kết đơn

Ở phần 1 và phần 2 ta đã biết cách xóa đi phần tử đầu (pHead) hoặc phần tử cuối (pTail) trong danh sách liên kết đơn. Trong một số trường hợp ta sẽ cần xóa đi một node có một khóa bất kỳ trong danh sách (khóa ở đây ta có thể hiểu là một **data** nào đó của một node trong danh sách liên kết đơn)

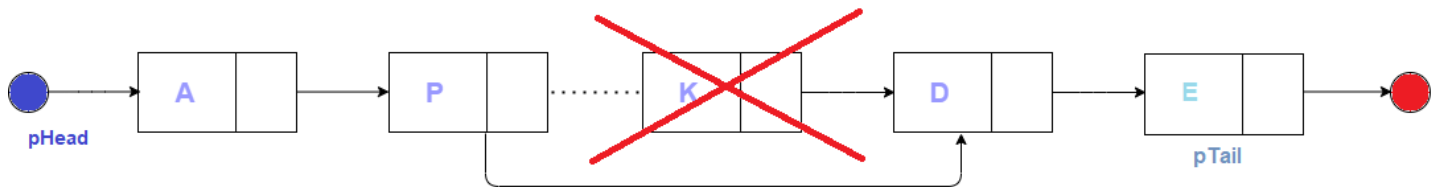
Để xóa một phần tử (hay một node) theo khóa k trong danh sách liên kết đơn, ta sẽ cần thực hiện 3 bước sau:

- **Bước đầu tiên:** Nhập một khóa k (hay có thể hiểu là **data**) vào để kiểm tra, tạo một **node p** và một **node k: $NODE *p, *k;$**
- **Bước thứ hai:** Duyệt toàn bộ danh sách từ đầu đến cuối danh sách thông qua node k
- **Bước thứ ba:** Nếu khóa k (hay **data**) nhập vào ở **bước 1** trùng với **data** của node đang duyệt thì gán: **$p \rightarrow next = k \rightarrow next$** có thể hiểu bước này là gán con trỏ next của node trước khóa k bằng con trỏ next của khóa k. Sau đó thực hiện xóa đi node k: ***delete k;***
- **Bước thứ tư:** Gán lại node p bằng node k: **$p = k;$**

Chú ý:

Nếu k (hay **data**) nhập vào bằng với **ds.pHead->data** thì gọi hàm xóa đầu **void XoaDau(ds)**, nếu k (hay **data**) nhập vào bằng với **ds.pTail->data** thì gọi hàm xóa cuối **void XoaCuoi(ds)**

Xem hình dưới đây để hiểu rõ hơn các bước trên:



Hàm **void XoaKhoak(LIST &ds, int k)** dưới đây thực hiện xóa phần tử có khóa k trong danh sách liên kết đơn. Hàm này cần truyền tham chiếu vào **LIST &ds** và **int data** chính là dữ liệu nhập vào để xóa đi node có khóa k trùng với **data** đó:

```

1. void XoaKhoak(LIST &ds, int data){
2.     //tao node p de luu tru phan tu truoc node k can xoa
3.     NODE *p = new NODE;
4.     //neu data dau vao bang voi pHead->data thi xoa dau
5.     if(ds.pHead->data == data){
6.         //goi ham xoa dau
7.         XoaDau(ds);
8.         //ket thuc ham
9.         return;
10.    }
11.    //neu data bang voi pTail->data thi xoa cuoi
12.    if(ds.pTail->data == data){
13.        //goi ham xoa cuoi
14.        XoaCuoi(ds);
15.        //ket thuc ham
16.        return;
17.    }
18.    //duyet qua cac phan tu co trong danh sach
19.    for(NODE *k = ds.pHead; k != NULL; k=k->next){
20.        //neu tim thay data trung voi k->data dang duyet
21.        if(k->data == data){
22.            //gan con tro next cua node p bang con tro next cua node k
23.            p->next = k->next;
24.            //xoa di node k
25.            delete k;
26.            //ket thuc ham
27.            return;
28.        }
29.        //gan node p bang node k de node p luon la node dung truoc node k
30.        p = k;
31.    }
32. }
```

4. Hủy toàn bộ danh sách liên kết đơn

Với 3 cách xóa ở trên, ta chỉ xóa đi một phần tử trong một lần chạy chương trình. Giả sử ta không cần sử dụng danh sách liên kết đơn nữa và ta cần xóa đi tất cả các node trong danh sách ta sẽ thực hiện hủy từng node có trong danh sách liên kết đơn đó.

Để thực hiện việc hủy danh sách, ta cần thực hiện 3 bước:

- **Bước đầu tiên:** Duyệt toàn bộ danh sách từ đầu đến cuối danh sách

- **Bước thứ hai:** Tạo một **node p** và gán bằng node đầu danh (pHead) sách: **$p = ds.pHead$** ;
- **Bước thứ ba:** gán phần tử đầu danh sách bằng node p trở đến next: **$ds.pHead = p \rightarrow next$**
- **Bước thứ tư:** Xóa đi **node p** ở bước 2: ***delete p***; và gán phần tử cuối danh sách bằng NULL: **$ds.pTail = NULL$** ;

Hàm **void HuyDS**(*LIST &ds*) dưới đây thực hiện việc xóa toàn bộ các node có trong danh sách liên kết đơn.

```

1. void HuyDS(LIST &ds){
2.     //duyet toan bo danh sach
3.     for(NODE *k = ds.pHead; k != NULL; k = k ->next)
4.     {
5.         //tao node p gan bang phan tu dau danh sach
6.         Node *p = ds.pHead;
7.         //gan phan tu dau danh sach bang p->next
8.         ds.pHead = p->next;
9.         //xoa di node p
10.        delete p;
11.    }
12.    //gan phan tu cuoi danh sach ve NULL
13.    ds.pTail = NULL;
14. }

```

📅 25/11/2021 (<https://laptrinhtudau.com/2021/11/25>)

🔗 danh sách liên kết đơn (<https://laptrinhtudau.com/tag/danh-sach-lien-ket-don/>) , delete list (<https://laptrinhtudau.com/tag/delete-list/>) , Single linked list (<https://laptrinhtudau.com/tag/single-linked-list/>)

📁 3 Danh Sách Liên Kết Đơn (<https://laptrinhtudau.com/cau-truc-du-lieu-giai-thuat/3-danh-sach-lien-ket-don/>)

f Facebook (<https://www.facebook.com/sharer/sharer.php?u=https://laptrinh tudau.com/xoa-phan-tu-danh-sach-lien-ket-don/>)

Twitter (<https://twitter.com/intent/tweet?text=Xóa%20phần%20tử%20trong%20danh%20sách%20liên%20kết%20đơn%20C/C++&url=https://laptrinhthudau.com/xoa-phan-tu-danh-sach-lien-ket-don/&via=Crunchify>)

in <https://www.linkedin.com/shareArticle?mini=true&url=https://laptrinh tudau.com/xoa-phan-tu-danh-sach-lien-ket-don/>

📌 Pinterest (<https://pinterest.com/pin/create/button/?ui>

```
>next%20=%20NULL;%20%20%20%20%20//xor%20node%20C
```

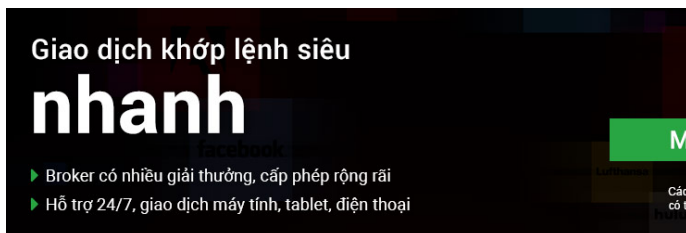
>next%20=%20NULL;%20%20%20%20%20%20%20%20%20%20%20%20//thay%20doi%20lai%20phan%20tu%20cuoi%20pTail%20cua%20danh%20sach%20
>next%20=%20k->next/em/strong%20có%20thể%20hiệu%20bước%20
language
>next;%20%20%20%20%20%20%20%20%20%20%20%20//xoá%20di%20nod

Thẻ nhớ microSD
Sandisk Ultra 100MBs
64GB - Hàng Nhập
Khẩu

QC Tiki

Tham Khảo Thêm

- Bài tập danh sách liên kết đơn bằng C/C++ (<https://laptrinh tudau.com/bai-tap-danh-sach-lien-ket-don-bang-c-c/>)
- Đảo ngược danh sách liên kết đơn C/C++ (<https://laptrinh tudau.com/dao-nguoc-danh-sach-lien-ket-don-c-c/>)
- Tìm kiếm phần tử trong danh sách liên kết đơn C/C++ (<https://laptrinh tudau.com/tim-kiem-phan-tu-trong-danh-sach-lien-ket-don-c-c/>)
- Nhập xuất trong danh sách liên kết đơn C/C++ (<https://laptrinh tudau.com/nhap-xuat-trong-danh-sach-lien-ket-don-c-c/>)
- Thêm phần tử vào giữa danh sách liên kết đơn C/C++ (<https://laptrinh tudau.com/them-phan-tu-vao-giua-danh-sach-lien-ket-don-c-c/>)
- Thêm phần tử vào cuối danh sách liên kết đơn C/C++ (<https://laptrinh tudau.com/them-phan-tu-vao-cuoi-danh-sach-lien-ket-don/>)
- Thêm phần tử vào đầu danh sách liên kết đơn C/C++ (<https://laptrinh tudau.com/them-phan-tu-vao-dau-danh-sach-lien-ket-don-c-c/>)
- Danh sách liên kết đơn – Cài đặt danh sách liên kết đơn C/C++ – Single linked list (<https://laptrinh tudau.com/danh-sach-lien-ket-don/>)













Bài Viết Mới



- | | |
|--|--|
| <ul style="list-style-type: none"> ■ Tìm kiếm nhị phân (Binary Search) trong C/C++ (https://laptrinh tudau.com/tim-kiem-nhi-phan-binary-search-trong-c/) ■ Tìm kiếm tuyến tính (Linear Search) trong C/C++ (https://laptrinh tudau.com/tim-kiem-tuyen-tinh-linear-search-trong-c/) ■ Hàm natsort() trong PHP (https://laptrinh tudau.com/ham-natsort-trong-php/) ■ Hàm natcasesort() trong PHP (https://laptrinh tudau.com/ham-natcasesort-trong-php/) | <ul style="list-style-type: none"> ■ Biến và kiểu dữ liệu trong ngôn ngữ lập trình C# (https://laptrinh tudau.com/bien-va-kieu-du-lieu-trong-ngon-ngu-lap-trinh-c/) ■ Hàm list() trong PHP (https://laptrinh tudau.com/ham-list-trong-php/) ■ Hàm ksort() trong PHP (https://laptrinh tudau.com/ham-ksort-trong-php/) ■ Ngôn ngữ lập trình C# (https://laptrinh tudau.com/c-sharp/) ■ Hàm krsort() trong PHP (https://laptrinh tudau.com/ham-krsort-trong-php/) |
|--|--|

- Bài tập tìm kiếm một nút trong cây nhị phân tìm kiếm C (<https://laptrinh tudau.com/bai-tap-tim-kiem-mot-nut-trong-cay-nhi-phan-tim-kiem-c/>)
- Hàm key() trong PHP (<https://laptrinh tudau.com/ham-key-trong-php/>)
- Hàm in_array() trong PHP (<https://laptrinh tudau.com/ham-in-array-trong-php/>)

Laptrinh tudau.com (<https://laptrinh tudau.com/>)

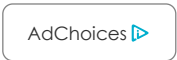
- | | |
|---|---|
|  Ngôn Ngữ SQL (https://laptrinh tudau.com/sql/) |  Ngôn Ngữ HTML (https://laptrinh tudau.com/html/) |
|  Ngôn Ngữ Python (https://laptrinh tudau.com/python/) |  Ngôn Ngữ CSS (https://laptrinh tudau.com/css/) |
|  Ngôn Ngữ PHP (https://laptrinh tudau.com/php/) |  Ngôn Ngữ C++ (https://laptrinh tudau.com/c++/) |
|  Ngôn Ngữ JavaScript (https://laptrinh tudau.com/javascript/) |  Ngôn Ngữ C (https://laptrinh tudau.com/lap-trinh-c/) |
|  Ngôn Ngữ Java (https://laptrinh tudau.com/java/) |  Cấu Trúc Dữ Liệu & Giải Thuật (https://laptrinh tudau.com/cau-truc-du-lieu-va-giai-thuat/) |

Kết Nối

-  (<https://www.facebook.com/laptrinh tudau>)  (<https://www.youtube.com/channel/UCIU0Su5svXlGd-dkW6YxlXw>)

Học Lập Trình

- 1 Lập Trình C (<https://laptrinh tudau.com/lap-trinh-c/>)
- 2 Lập Trình C++ (<https://laptrinh tudau.com/c++/>)
- 3 Cấu Trúc Dữ Liệu & Giải Thuật (<https://laptrinh tudau.com/cau-truc-du-lieu-va-giai-thuat/>)
- 4 Lập Trình Python (<https://laptrinh tudau.com/python/>)
- 5 Lập Trình Java (<https://laptrinh tudau.com/java/>)
- 6 Lập Trình SQL (<https://laptrinh tudau.com/sql/>)
- 7 Lập Trình PHP (<https://laptrinh tudau.com/php/>)
- 8 Lập Trình Javascript (<https://laptrinh tudau.com/javascript/>)



LAPTRINHTUDAU.COM

KẾT NỐI

LIÊN HỆ


ỦNG HỘ BLOG


Laptrinhthudau.com – một website chuyên về môn học lập trình C/C++ dành cho người mới bắt đầu học lập trình. Nơi đến nâng cao, dễ dàng tiếp cận cho những người mới bắt đầu nhập môn học lập trình!


📧 Email: laptrinhthudau@gmail.com | [Facebook \(https://www.facebook.com/laptrinhthudau\)](https://www.facebook.com/laptrinhthudau) | [YouTube \(https://www.youtube.com/channel/UCv0dUy5sUgCHbaU7KhaG4Dn6M\)](https://www.youtube.com/channel/UCv0dUy5sUgCHbaU7KhaG4Dn6M)


📱 Zalo: <https://www.zalo.me/laptrinhthudau> | [Instagram \(https://www.instagram.com/laptrinhthudau\)](https://www.instagram.com/laptrinhthudau)

Ngân Hàng: MB BANK^

 Youtube (<https://www.youtube.com/channel/UCdIg4nh0yP0G4sm9YJW5n79962745>)

 Facebook (<https://www.facebook.com/groups/laptrinh-tu-dau>)

 TikTok (https://www.tiktok.com/@laptrinh_tudau)

 Telegram (https://www.telegram.com/laptrinh_tudau)

Số Tài Khoản: 1110110246810

Ngân Hàng: BIDV

Số Tài Khoản: 2151000325392