

لیست کاغذی

توجه کنید صرفاً با ذکر دلیل به سوالات پاسخ دهید و از ارائه پاسخ های کوتاه خودداری کنید

(1) آرایه را با لیست پیوندی مقایسه کنید. تفاوت های این دو ساختمان داده را مشخص کنید.

(2) از شما دعوت شده برای هر یک از مسئله های زیر ساختمان داده درست را بین آرایه و لیست پیوندی یا ترکیب های آن مشخص کنید و دلیل آنرا شرح دهید.

- در برنامه ای شما مجبور شده اید حجم زیادی حدود ۱۰ به توان ۶ عنصر و یا بیشتر را در رم خود ذخیره کنید با ذکر دلیل بگویید کدام ساختمان داده میتواند گزینه مناسبی برای ذخیره این نواع دیتا باشد .
- در برنامه شما باید اطلاعات حدود ۱۰ به توان ۳ مدل غذای مختلف را حداکثر ذخیره کنید و شما هر روز باید قیمت هر غذا رو بروزرسانی کنید بهترین روش ذخیره سازی غذا چگونه است؟
- در نرم افزار شما تمام فروشگاه های اسباب بازی شهر باصفای همدان ثبت شده اند و احتمال تعطیل شدن یا اضافه شدن فروشگاه ها به دلیل ثبات اقتصادی کم است اما ناممکن نیست. هر فروشگاه تعداد نامعلومی از اقلام دارد که مدام اضافه و کم میشوند بهترین روش برای ذخیره سازی هر کدام از اقلام فروشگاه چیست؟

(3) مزایا و معایب استفاده از لیست پیوندی در مقایسه با آرایه داینامیک (Dynamic Array) چیست؟

(4) تفاوت های اصلی بین لیست پیوندی خطی و لیست پیوندی دایره ای چیست؟ مزایا و معایب هر کدام چیست؟

یه مسئله کاغذی دیگه

توضیحات : در این سری از تمرین شما باید ابتدا الگوریتم حل مسئله مربوطه را شرح دهید سپس شبه کدی خوانا و صحیح به همراه مثالی ساده و عددی برای درک بیشتر ارائه دهید. توجه کنید ارسال شبه کد به همراه الگوریتم حل مسئله مورد نظر هست و از ارسال جواب کوتاه و توضیح الگوریتم بدون شبه کد خودداری کنید.

1) برنامه ای بنویسید که یک عنصر از لیست را حذف کند.

2) الگوریتمی بنویسید که دو لیست پیوندی جداگانه را با هم متصل کند.

3) برنامه ای بنویسید گره تکراری را در لیست پیوندی پیدا و حذف کند (در صورت وجود).

4) برنامه ای بنویسید که وارون یک لیست پیوندی را در خروجی چاپ کند. (منظور این است لیست پیوندی معکوس شود)

پیاده سازی لیست پیوندی

یک چارچوب از نحوه پیاده سازی یک لیست پیوندی برای کمک به شما عزیزان داده شده است. شما باید این چهارچوب پیاده سازی را کامل کنید. توجه کنید نیازی به ارائه مثال ندارید صرفا خواسته شما سنجش درکتون از نحوه پیاده سازی این ساختار داده هست.

```
1 struct Node {
2
3     int    val    {};
4     Node* next {};
5
6 };
7
8 class LinkList{
9
10 public:
11     //insert a node with value = val
12     void insert(int val);
13
14     //delete all nodes with value = val
15     void deleteNode(int val);
16
17     //count all nodes that value = val
18     int count(int val);
19
20     //find first nodes that value = val
21     Node* find(int val);
22
23     //return first node of link list
24     Node* front();
25
26     //return last node of link list
27     Node* back();
28
29 private:
30
31     Node* head;
```

```
32 |  
33 | };
```

لیست پیوندی دو طرفه دایره ای

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

مانند سوال قبل یک چارچوب از نحوه پیاده سازی یک لیست پیوندی دو طرفه دایره ای برای کمک به شما عزیزان داده شده است. شما باید این چهارچوب پیاده سازی را کامل کنید. توجه کنید نیازی به ارائه مثال ندارید صرفا خواسته شما سنجش درکتون از نحوه پیاده سازی این ساختار داده هست.

```
1 struct Node {
2     int val{};
3     Node* next{};
4     Node* last{};
5 };
6
7 class LinkList2{
8
9 public:
10
11     //insert a node with value = val
12     void insert (int val);
13
14     //delete all nodes with value = val
15     void deleteNode(int val);
16
17     //print all nodes
18     void print();
19
20 private:
21     Node* head;
22 };
```

گره میانی در دسر ساز

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

کوروش دانشجوی نمونه درس ساختمان داده هست. به او سوالی داده شده که باید میانه یک لیست پیوندی را پیدا کند. حال برای حل سوال از شما کمک گرفته است که تابعی بنویسید که میانه یک لیست پیوندی پیدا کند. دقت کنید اگر تعداد ورودی ها زوج بود دومین گره میانی را به عنوان میانه در نظر میگیرم.

مثال

ورودی نمونه ۱

12345

خروجی

3

مثال

ورودی نمونه 2

123456

خروجی

مجموع زوج ها

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

کوروش داستان ما یک سوال خوب دیگه بهش داده شده ولی چون دیگه حوصله حل سوال را ندارد از شما میخواهد که به سوالاتش پاسخ دهید . او قرار هست لیست پیوندی ای داشته باشد که شامل تعدادی عدد با یک فاصله باشد (بسته به نیاز کاربر) حال او قرار هست گره های زوج این لیست پیوندی ما را با هم جمع بزند . به او کمک کنید که بتواند این کار را انجام دهد تذکر : منظور از گروه زوج درواقع اندیس زوج اون گره هست نه اینکه اعداد زوج رو با هم جمع بزنید!

مثال

برای مثال در نمونه زیر گره های 2و4و6 زوج اند که مجموع آن ها 12 میشود

ورودی نمونه ۱

1 20 3 30 5 50

ورودی خروجی۱

100

درواقع اعداد ایندکس های شماره 2و4و6 با هم جمع شدند یعنی : $100 = 50 + 30 + 20$

لیست پیوندی مقلوب

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

داریوش که برادر بزرگتر کوروش هست به حل سوالات خلاقانه تر علاقه زیادی دارد. او میخواد بررسی کند که آیا لیست پیوندی که کاربر به او میدهد خاصیت Palindrome یا همان برابر بودن نیمه اول عدد با نیمه دوم عدد را دارد یا خیر برای مثال ما لیست پیوندی به ما داده شده و میخواهیم بفهمیم اگر آن را به دو نیم تقسیم کنیم آیا نیمه سمت چپ با نیمه سمت راست برابر است یا خیر .

مثال

ورودی نمونه 1

12321

خروجی نمونه 1

True

اجتماع و اشتراک

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

با توجه به دو لیست پیوندی یک طرفه دلخواه که به صورت Hard Code ساخته میشود از شما می‌خواهیم اجتماع و اشتراک عناصر موجود در دو لیست پیوندی را، به صورت دو لیست پیوندی مختلف A, B نمایش دهد

****** حال دو نمونه ورودی و خروجی برای راحتی شما تدارک دیده ایم تا منظور سوال رو بهتر و دقیق تر متوجه بشید . دقت کنید این سوال به صورت Hard Code است و ورودی از کاربر گرفته نمیشود صرفاً برای دید شما از این سوال یک ورودی فرضی در نظر گرفته شده است.

ورودی

ورودی شامل دو خط است که خط اول، شامل لیست پیوندی اول می‌باشد و در خط دوم، لیست پیوندی دوم نیز قرار دارد که به صورت Hard Code مقدار دهی اولیه می‌شوند (توضیحات بیشتر در پیوند)

خروجی

خروجی برنامه‌ی شما باید شامل ۲ لیست پیوندی باشد که در خط اول لیست پیوندی اجتماع عناصر به نمایش داده می‌شود و در خط دوم لیست پیوندی شامل اشتراک دو لیست داده شده نشان داده می‌شود.

مثال

ورودی نمونه ۱

10 15 4 20
8 4 2 10

خروجی نمونه ۱

10 15 4 20 8 2
4 10

کارفرمای نگران

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

کیارش، یک برنامه‌نویس باهوش و خلاق، همیشه به چالش‌های جدید علاقه داشت. یک روز، وقتی که در دفتر کارش نشسته بود، رئیسش با چهره‌ای نگران به او نزدیک شد و گفت:

"کیارش! لیست اطلاعات مشتری‌ها کاملاً به هم ریخته است. باید آن را هرچه سریع‌تر مرتب کنی!"

کیارش به صفحه‌ی مانیتورش نگاه کرد. لیست بلندی از داده‌ها مقابلش بود، اما مشکل اینجا بود که این لیست یک **لیست پیوندی** (Linked List) بود، نه یک آرایه معمولی! این یعنی که نمی‌توانست به‌سادگی با یک تابع سورت معمولی، همه چیز را مرتب کند.

او نفس عمیقی کشید و با خودش گفت: **"پس باید با الگوریتم‌های مرتب‌سازی مخصوص لینک لیست‌ها کار کنم."**

با دقت، او الگوریتم مورد نیازش را طراحی کرد. قدم به قدم، اشاره‌گرها را حرکت داد، لیست را به بخش‌های کوچک‌تر تقسیم کرد و دوباره آن‌ها را به شکل مرتب‌شده به هم متصل کرد.

پس از چند ساعت کدنویسی و تست، بالاخره لیست مرتب شد! رئیسش که نتیجه را دید، لبخندی زد و گفت: **"مثل همیشه عالی بود کیارش! لیست مثل روز اولش شده!"**

کیارش لبخند زد. او یک چالش دیگر را با موفقیت پشت سر گذاشته بود. اما خوب می‌دانست که دنیای برنامه‌نویسی همیشه پر از لیست‌های به‌هم‌ریخته‌ی جدید است که باید مرتبشان کند!

مثال

در اینجا چند نمونه برای فهم بهتر صورت سوال و قالب ورودی و خروجی تست‌ها داده می‌شود.

ورودی نمونه ۱

1 2 3 4 5

خروجی نمونه ۱

1 2 3 4 5

ورودی نمونه ۲

9 7 5 3 1

خروجی نمونه ۲

1 3 5 7 9