Dynamic Programming | Set 20 (Maximum Length Chain of Pairs)

You are given n pairs of numbers. In every pair, the first number is always smaller than the second number. A pair (c, d) can follow another pair (a, b) if b < c. Chain of pairs can be formed in this fashion. Find the longest chain which can be formed from a given set of pairs.
Source: Amazon Interview | Set 2

For example, if the given pairs are {{5, 24}, {39, 60}, {15, 28}, {27, 40}, {50, 90} }, then the longest chain that can be formed is of length 3, and the chain is {{5, 24}, {27, 40}, {50, 90}}

This problem is a variation of standard Longest Increasing Subsequence problem. Following is a simple two step process.
1) Sort given pairs in increasing order of first (or smaller) element.
2) Now run a modified LIS process where we compare the second element of already finalized LIS with the first element of new LIS being constructed.

The following code is a slight modification of method 2 of this post.

```
#include<stdio.h>
#include<stdlib.h>

// Structure for a pair
struct pair
{
  int a;
  int b;
};

// This function assumes that arr[] is sorted in increasing order
// according the first (or smaller) values in pairs.
```


```
int maxChainLength( struct pair arr[], int n)
{
   int i, j, max = 0;
   int *mcl = (int*) malloc ( sizeof( int ) * n );

   /* Initialize MCL (max chain length) values for all indexes */
   for ( i = 0; i < n; i++ )
      mcl[i] = 1;

   /* Compute optimized chain length values in bottom up manner */
   for ( i = 1; i < n; i++ )
      for ( j = 0; j < i; j++ )
         if ( arr[i].a > arr[j].b && mcl[i] < mcl[j] + 1)
            mcl[i] = mcl[j] + 1;

   // mcl[i] now stores the maximum chain length ending with pair i

   /* Pick maximum of all MCL values */
   for ( i = 0; i < n; i++ )
      if ( max < mcl[i] )
         max = mcl[i];

   /* Free memory to avoid memory leak */
   free( mcl );

   return max;
}


/* Driver program to test above function */
int main()
{
    struct pair arr[] = { {5, 24}, {15, 25},
                          {27, 40}, {50, 60} };
    int n = sizeof(arr)/sizeof(arr[0]);
    printf("Length of maximum size chain is %d\n",
           maxChainLength( arr, n ));
    return 0;
}
```

Output:

```
Length of maximum size chain is 3
```

Time Complexity: O(n^2) where n is the number of pairs.

Geometric Algorithms

Popular Posts

All permutations of a given string
Memory Layout of C Programs
Understanding “extern” keyword in C
Median of two sorted arrays
Tree traversal without recursion and without stack!
Structure Member Alignment, Padding and Data Packing
Intersection point of two Linked Lists
Lowest Common Ancestor in a BST.
Check if a binary tree is BST or not
Sorted Linked List to Balanced BST

The given problem is also a variation of Activity Selection problem and can be solved in (nLogn) time. To solve it as a activity selection problem, consider the first element of a pair as start time in activity selection problem, and the second element of pair as end time. Thanks to Palash for suggesting this approach.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Related Tpoics:

- Remove minimum elements from either side such that 2*min becomes more than max
- Divide and Conquer | Set 6 (Search in a Row-wise and Column-wise Sorted 2D Array)
- Bucket Sort
- Kth smallest element in a row-wise and column-wise sorted 2D array | Set 1
- Find the number of zeroes
- Find if there is a subarray with 0 sum
- Divide and Conquer | Set 5 (Strassen’s Matrix Multiplication)
- Count all possible groups of size 2 or 3 that have sum as multiple of 3

Writing code in comment? Please use **ideone.com** and share the link here.

23 Comments GeeksforGeeks

Sort by Newest

Join the discussion…

AlienOnEarth • 16 days ago

Algorithm for Greedy Approach:

1.) Sort the array based on last element (Finish Time)
2.) Apply O(o^2) Algorithm to find all the pairs which satisfy Pairing property. W
count
3.) return max pair count

Reply • Share ›

smith • 5 months ago

what is need of sorting first

1 Reply • Share ›

Shiwakant Bharti • 9 months ago

Hi Omar/All,
I have attached Java solution for the same. Please have a look and comment.

1 Reply • Share ›

shiwakant.bharti • 9 months ago

Here is Java adaptation as requested by Omar.
Note: This is just contents which can be included inside any class.

```
static class Pair implements Comparable<Pair> {
```

Recent Comments

Aman Hi, Why arent we checking for conditions...

Write a C program to Delete a Tree. · 6 minutes ago

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 10 minutes ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 35 minutes ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

Count trailing zeroes in factorial of a number · 37 minutes ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 1 hour ago

newCoder3006 Code without using while loop. We can do it...

Find subarray with given sum · 1 hour ago

```
    /**
     * @param a
     * @param b
     */
    public Pair(int a, int b) {
        super();
        this.a = a;
        this.b = b;
    }

    public int a, b;

    /*
     * (non-Javadoc)
```

see more

• Reply • Share ›

shiwakant.bharti shiwakant.bharti • 9 months ago

Admin: Looks like there is some formatting problem here, kindly copy it
comments. I used the language="Java" do I need type in "java" or "JAV

• Reply • Share ›

Krum Bakalsky • 9 months ago

I think that this is simply solved in O(n) time, because it is an instance of the lo
can be solved in O(V + E) time.

• Reply • Share ›

Silent • 9 months ago

i guess we can simply do it using greedy approach by sorting int the increasing
fast??

gargsanjay • 11 months ago

after sorting....
step 1) pick the pair with smallest b value.
step 2)among the pairs with a value greater than b value of previous node in m
what abt this algorithm..

Omar Hidayat • a year ago

how to convert in code pair to java.

abhishek08aug • a year ago

Intelligent :D

```
/* Paste your code here (You may delete these lines if not writing c
```


Napster • a year ago

Dont you think ,with activity selection algo,order of pairs would be depleted.

```
/* Paste your code here (You may delete these lines if not writing c
```


Arun • a year ago

I believe that the above question can be easily solved using this approach:

1. Sort the pairs based in the increasing order of second values, e.g. (5,24), (1

2. Now add the first pair to the list.

3. Start with i = 0; and check if y(i) < x(i+1), then add x(i+1)(i+1) to the list, else

4. Continue iterating over the list till i == size of the given pairs....

Done !!!

1 ^ | v • Reply • Share ›

makeit ➦ Arun • a year ago

What if the given pairs are (1, 100),5,24), (15, 28), (27, 40), (39, 60) an

1.sort the list,
2. add the first one first.
3. then the result is 1,
4. not good

or else:
for each pair in the list, consider it as the first pair in the list.
In this case, the overall run time is still o(n**2)

```
/* Paste your code here (You may delete these lines if not wri
```

^ | v • Reply • Share ›

yesminister ➦ Arun • a year ago

Arun: You are right. In fact that's exactly the method employed in the ac
explanation above.

^ | v • Reply • Share ›

VikasG ➦ Arun • a year ago

I think you have misunderstood the problem. You are not allowed to ch
By sorting, you are solving the problem for a different input.

^ | v • Reply • Share ›

rahul • 2 years ago

Why sorting??

```
/* Paste your code here (You may delete these lines if not writing co
```

2 ^ | v • Reply • Share ›

zoe ➦ rahul • a year ago

Yeah can't understand why we need sorting here...
Can be done using LIS method exactly without sorting...

```
/* Paste your code here (You may delete these lines if not writ
```

^ | v • Reply • Share ›

Zavatar ➦ zoe • 8 months ago

It confused me too at first.

But then I realized the result is constructed from a given set of

^ | v • Reply • Share ›

Palash • 2 years ago

Isn't this the standard job scheduling problem and can be done in NlogN by gre
with nearest end time at each step ?

1 ^ | v • Reply • Share ›

Abhinav ➦ Palash • a year ago

Exactly!
Thats what I thought!

```
/* Paste your code here (You may delete these lines if not writ
```

Reply • Share ›

kartik ➦ Palash • 2 years ago

@Paiash: Thanks for sharing your thoughts. This in fact looks similar t

Reply • Share ›

Palash • 2 years ago

Isn't this, job scheduling problem, and can be done in nlogn, by greedy ?

```
/* Paste your code here (You may delete these lines if not writing co
```

Reply • Share ›

anonymous • 2 years ago

is it not simple?

just sort them based on the second values

pick the next one which has first value greater than the current one's second v

```
/* Paste your code here (You may delete these lines if not writing co
```

Reply • Share ›

Subscribe Add Disqus to your site