Χ





reviewer4@nptel.iitm.ac.in ~

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Programming, Data Structures And Algorithms

Using Python (course)

Announcements (announcements)

About the Course (https://swayam.gov.in/nd1_noc19_cs40/preview) Ask a Question (forum)

Progress (student/home) Mentor (student/mentor)

Course outline

How to access the portal

Week 1: Introduction

Week 1 Quiz

Week 2: Basics of Python

Week 2 Quiz

Week 2 Programming Assignment

Week 3: Lists, inductive function definitions, sorting

Week 3 Programming Assignment

Week 8 Programming Assignment

Due on 2019-09-27, 23:59 IST

For this assignment, you have to write a complete Python program. Paste your code in the window below.

- You may define additional auxiliary functions as needed.
- There are some public test cases and some (hidden) private test cases.
- "Compile and run" will evaluate your submission against the public test cases
- "Submit" will evaluate your submission against the hidden private test cases. There are 10 private test cases, with equal weightage. You will get feedback about which private test cases pass or fail, though you cannot see the actual test cases.
- Ignore warnings about "Presentation errors".

Domino Solitaire

(Indian National Olympiad in Informatics, 2008)

In Domino Solitaire, you have a grid with two rows and many columns. Each square in the grid contains an integer. You are given a supply of rectangular 2 × 1 tiles, each of which exactly covers two adjacent squares of the grid. You have to place tiles to cover all the squares in the grid such that each tile covers two squares and no pair of tiles overlap.

The score for a tile is the difference between the bigger and the smaller number that are covered by the tile. The aim of the game is to maximize the sum of the scores of all the tiles.

Here is an example of a grid, along with two different tilings and their scores.

Week 4: Sorting, Tuples, Dictionaries, Passing **Functions, List** Comprehension

Week 4 Quiz

Week 4 **Programming Assignment**

Week 5: **Exception** handling, input/output, file handling, string processing

Week 5 **Programming Assignment**

Week 6: Backtracking, scope, data structures; stacks, queues and heaps

Week 6 Quiz

Week 7: Classes, objects and user defined datatypes

Week 7 Quiz

Week 8: **Dynamic** programming, wrap-up

Week 8 **Programming Assignment**

Week 8 **Programming Assignment** (/noc19_cs40/progassignment? name=98)



	Tiling 1					
	8	6	2	3		
	9	7	1	2		
C 10						

Tiling 2						
8	6	2	3			
9	7	1	2			
Score 6						

The score for Tiling 1 is 12 = (9-8)+(6-2)+(7-1)+(3-2) while the score for Tiling 2 is 6 = (8-6)+(9-7)+(3-2)+(2-1). There are other tilings possible for this grid. but you can check that Tiling 1 has the maximum score among all tilings.

Your task is to read the grid of numbers and compute the maximum score that can be achieved by any tiling of the grid.

Solution hint

Recursively find the best tiling, from left to right. You can start the tiling with one vertical tile or two horizontal tiles. Use dynamic programming to evaluate the recursive expression efficiently.

Input format

The first line contains one integer N, the number of columns in the grid. This is followed by 2 lines describing the grid. Each of these lines consists of N integers, separated by blanks.

Output format

A single integer indicating the maximum score that can be achieved by any tiling of the given grid.

Test Data:

For all inputs, $1 \le N \le 105$. Each integer in the grid is in the range $\{0,1,...,10^4\}$.

Sample Input:

4 8 6 2 3 9 7 1 2

Sample Output:

12

Sample Test Cases

Output

Download videos

Text Transcripts

Online Programming Test - Sample

Online Programming Test 1, 26 Sep 2019, 09:30-11:30

Online Programming Test 2, 26 Sep 2019, 20:00-22:00 Test 4
Case 8 6 2 3
1 9 7 1 2

9 7 1 2

10

Test 8789 7959 4809 5257 4592 9455 6462 5855 6399 95
Case 69
2 4977 5499 7329 2997 9599 5445 2412 9838 6252 65

4977 5499 7329 2997 9599 5445 2412 9838 6252 65

100

2511 2090 9410 4226 3959 3826 2318 5356 5333 86
30 9624 3155 7360 6547 503 4539 8065 6558 8119
8299 792 2046 6803 6519 9765 851 2039 2315 143
1566 141 7040 894 5713 9574 2861 1437 8254 8573
3503 2540 2862 8272 5518 9578 155 8493 9935 167
2 5874 5457 3379 3689 6102 9972 4269 3263 274 8
535 2766 1393 1859 2864 8412 368 6360 9530 1607
5327 6394 6831 86 7476 1983 1257 9508 5275 8492
8620 4276 800 5409 2229 6220 8377 2016 1569 125
5 1554 4253 3592 8325 8073 4123 5605 7625 4737
5013 4173 2287

Test Case

3

9668 4457 791 6609 6438 9208 9074 5723 6687 494
0 3855 3866 7280 6290 3158 7736 7585 9150 5101
5567 8238 605 3218 3442 6767 7493 2552 6121 780
3 9479 1702 7483 7379 9357 1309 4021 6197 2206
402 6193 5867 6284 8661 5558 3199 5171 4723 838
8 9933 827 9738 7870 1030 6640 7850 249 2164 41
76 4203 4686 2685 5869 9403 698 1360 1954 1818
464 9144 5064 5033 9785 2402 1599 3597 1153 594
2 9486 1823 4149 3317 6659 5671 2763 753 518 93
01 399 5176 3041 5035 2088 8825 8874 7437 9378
6412 9721 9874 7499

Test Case

1000

9732 3404 5420 3913 7633 7535 876 366 4670 9237 6093 9707 7672 2047 9543 5470 4593 3758 6787 55 18 6895 560 8951 28 6474 2192 2293 8463 6631 22 26 8236 1160 7880 5644 8338 3952 2066 8659 1334 1098 7758 9694 3964 1420 7548 2242 5258 9954 32 49 9058 6006 6735 8491 7324 4498 820 3374 7984 3460 715 2218 1681 6188 7647 679 3398 6083 331 1034 8101 9192 4898 6143 7990 4630 1582 925 630 7 447 6808 1033 3390 9820 781 8331 3529 3529 32 24 1747 9415 3334 4048 5649 8040 7716 8457 154 3918 6929 3176 3359 3654 593 9382 2469 9232 509 0 9717 2949 4831 3648 9075 4250 5399 2581 8122 178 1371 8244 2693 7789 7447 679 3086 2668 8896 8825 3019 832 3779 4953 5869 9247 6313 1333 411 3 2355 5484 3026 9279 1082 4309 7390 2846 1960 1760 2900 1269 6188 9505 9004 9756 4314 5033 11 85 2638 9624 7411 4327 2610 9052 364 5821 5215

43

08

24

9

42

54

23

12

31

59

7

https://onlinecourses.nptel.ac.in/noc19_cs40/progassignment?name=98

Test Case 5

10000

1940 6546 5520 785 4363 4624 6459 469 8045 5305 780 5228 786 7050 9939 4784 9874 5750 3468 7704 7922 5696 439 5110 3564 5600 4832 9937 1510 882 1 9182 2365 7743 4955 5979 4855 8353 2709 8937 7050 9939 8021 885 446 7248 3104 2905 144 9015 895 1490 291 6516 9535 9760 6330 4487 5085 2703 206 465 9899 9427 9995 3789 3779 1613 7571 8098 5171 2802 3013 2808 8057 8361 2328 7427 4448 87 18 2129 3364 3196 5727 6393 7875 8660 3135 2917 3676 1039 755 1275 1406 7213 8826 6768 4329 296 5 4487 2067 273 5622 8611 7109 2774 9465 5250 9 654 5568 3743 1579 1164 3952 6494 6820 1982 611 5 3585 9676 3959 3302 5867 5288 9525 1677 2922 1548 1999 1116 6785 7468 9733 7935 5879 1576 50 33 3933 8256 9270 138 3238 4551 4681 5722 9565 7658 4666 78 8743 6287 1661 8617 7157 8971 4534 6523 4229 9762 5572 1843 2715 5316 8 381 1621 9 206 6983 7172 3500 4356 6759 5830 7035 223 4812 6727 1786 1838 5081 6713 40 2529 9883 7485 9637 42905211

20

Test Case

6

1 3 9 19 9 11 13 15 21 31 21 23 25 27 29 35 45 35 37 39 2 4 10 20 10 12 14 16 22 32 22 24 26 28 30 36 4

6 36 38 40

11

0

100 1 7 17 7 9 11 13 15 17 19 21 23 25 27 29 35 45 35 37 39 41 47 57 47 53 63 53 55 57 59 61 67 77 71 81 71 73 75 77 79 81 83 89 99 89 91 97 107 9 7 99 101 103 105 107 113 123 113 115 117 119 12 1 123 125 127 129 135 145 135 137 139 141 143 1 49 159 149 151 153 155 161 171 161 163 165 167 169 175 185 175 177 179 181 187 197 187 189 191 193 195 197 199 2 8 18 8 10 12 14 16 18 20 22 24 26 28 30 36 46 36 38 40 42 48 58 48 54 64 54 56 58 60 62 68 78 72 82 72 74 76 78 80 82 84 90 100 90 92 98 108 98 100 102 104 106 108 114 124 114 116 118 120

122 124 126 128 130 136 146 136 138 140 142 144 150 160 150 152 154 156 162 172 162 164 166 168 170 176 186 176 178 180 182 188 198 188 190 192 50

8

Test Case 194 196 198 200

Test

Case

1000

53 06

Test Case

1000

1 3 5 7 9 11 13 15 17 19 21 23 25 31 41 31 33 3 5 37 39 41 43 45 47 53 63 53 59 69 63 73 63 69 79 69 71 73 75 77 79 81 83 85 87 93 103 93 95 9

40

46

Test Case 10

39 22

4

```
Test
Case
     8 6 2 3
                                                         12
     9 7 1 2
11
     10
Test
     8789 7959 4809 5257 4592 9455 6462 5855 6399 95
                                                         31
Case
                                                         59
12
     4977 5499 7329 2997 9599 5445 2412 9838 6252 65
                                                         7
     77
     100
     2511 2090 9410 4226 3959 3826 2318 5356 5333 86
     30 9624 3155 7360 6547 503 4539 8065 6558 8119
     8299 792 2046 6803 6519 9765 851 2039 2315 143
     1566 141 7040 894 5713 9574 2861 1437 8254 8573
     3503 2540 2862 8272 5518 9578 155 8493 9935 167
     2 5874 5457 3379 3689 6102 9972 4269 3263 274 8
     535 2766 1393 1859 2864 8412 368 6360 9530 1607
     5327 6394 6831 86 7476 1983 1257 9508 5275 8492
     8620 4276 800 5409 2229 6220 8377 2016 1569 125
Test
     5 1554 4253 3592 8325 8073 4123 5605 7625 4737
                                                         42
Case
    5013 4173 2287
                                                         54
13
     9668 4457 791 6609 6438 9208 9074 5723 6687 494
                                                         23
     0 3855 3866 7280 6290 3158 7736 7585 9150 5101
     5567 8238 605 3218 3442 6767 7493 2552 6121 780
     3 9479 1702 7483 7379 9357 1309 4021 6197 2206
     402 6193 5867 6284 8661 5558 3199 5171 4723 838
     8 9933 827 9738 7870 1030 6640 7850 249 2164 41
     76 4203 4686 2685 5869 9403 698 1360 1954 1818
     464 9144 5064 5033 9785 2402 1599 3597 1153 594
     2 9486 1823 4149 3317 6659 5671 2763 753 518 93
     01 399 5176 3041 5035 2088 8825 8874 7437 9378
     6412 9721 9874 7499
```

The due date for submitting this assignment has passed.

As per our records you have not submitted this assignment.

Sample solutions (Provided by instructor)

```
def solve(l1,l2):
 1
2
3
4
           n = len(11)
           if n == 0:
5
6
7
8
9
                  return(0)
           ans = [0 \text{ for } i \text{ in } range(n+1)]
           ans[n-1] = max(l1[n-1], l2[n-1]) - min(l1[n-1], l2[n-1])
           for i in range(n-2,-1,-1):
    vert = max(l1[i],l2[i]) - min(l1[i],l2[i]) + ans[i+1]
    horz = max(l1[i],l1[i+1]) - min(l1[i],l1[i+1]) + max(l2[i],l2[i])
    ans[i] = max(vert,horz)
11
12
13
14
<u>15</u>
16
17
           return(ans[0])
18
    nstr = input()
    # Value of n not needed in Python
# n = int(nstr.strip())
19
20
    # Read and parse first row of numbers
```

```
line1str = input().strip()
line1strlist = line1str.split()
line1 = []
for s in line1strlist:
        line1.append(int(s))

# Read and parse second row of numbers
line2str = input().strip()
line2strlist = line2str.split()
line2 = []
for s in line2strlist:
        line2.append(int(s))

print(solve(line1,line2))
```