

Programming Problems

```

,-----,
| Problem 0: Helloworld |
`-----'

```

The Problem

~~~~~

(Freebie) Write a program to print the sentence, Hello world on the screen. There is one space between the two words, and a newline after World. The program should then input an integer, add 10 to it and print the sum on the screen, followed by a newline. Note: You need to submit this program first to register yourself. .. 2 marks

### In Short

~~~~~

```

int main()
{
    int x;
    printf("Hello world\n");
    scanf("%d", &x);
    printf("%d\n", x+10);
}

```

Comments

~~~~~

This program tests the editor, compiler, submission script, mail, printf, scanf, +, and just about everything else. In addition, it registers the person by creating his program and score directories.

Some smart IITian tried out

```
int *x; scanf("%d", x); printf("%d\n", *x+10);
```

which promptly crashed, leaving him wondering why.

Someone else (actually, me) did "Hello World" (caps W) and got incorrect. My fault, actually. I forgot to lowercase it before checking.

```

,-----,
| Problem 1: Monotonic |
`-----'

```

### The Problem

~~~~~

An IITian student, having forgotten to attend his last Physics lab, has to now cook up a graph for the lab report. He has a moth-eaten journal from ten years ago of the same lab, in which the readings are there, but not in the same order that they were taken down from the experiments. Each reading has an x-value which runs from 1 from to the number of readings, and a y-value, which may be any positive integer. However, only the latter is visible for each

reading, in the old report.

The student knows that the final graph is a sine wave. With a couple of hours left for submission, he decides that the closest he can get to that is to plot a 'zigzag graph'. He will rearrange the readings in such a way that when a line is drawn between successive readings on the graph, a zigzag pattern is obtained, with every point being at a local maxima or minima of the graph. There may be more than one reading with the same y-value, but if they are placed adjacent, the lab instructor will spot the student's ploy immediately and have him repeat the experiment. Hence, he wants to avoid an arrangement in which this happens.

Write a program that given a sequence of readings, determines whether they suit the student's requirements or not. The input consists of $n+1$ lines, with a single integer on each, when there are n readings. The first number is the number of readings, i.e. n , and the remaining numbers are the y-values of the readings, given in no particular order. The output should be the word 'no' in lowercase letters, if the readings can't help the hapless student, or the required sequence, if they can. In the latter case, the readings, in the correct order, should be output as a single integer on each line. Hence, for an n -reading problem, for which a 'good' rearrangement exists, the answer will consist of n lines. There will be at most 500 readings, with y-values in $(0,10)$ 15 marks

In Short

~~~~~

We define a monontonic sequence as one which strictly increases and decreases alternately, starting with either increase or decrease. Given a sequence of numbers, find a monontonic permutation if you can, or print "no" if you can't.

My approach

~~~~~

If the number with the maximum number of occurrences occurs more than $N/2$ times, then it's not possible to find a sequence. Otherwise alternately output numbers on either side of this most frequent number. The problem would become more interesting if we insist that the final sequence has to have increasing magnitude.

Alternatively, (due to sami), one might use divide and conquer, knowing that an odd length sequence may 'have' to start with an increase or a decrease, but an even length sequence, on reversing, starts with the other kind of change. Therefore we can have a merge sort like algo with the merging step optionally reversing one or both the even-length (power of 2) sequences and concatenating.

Winner1 algo

~~~~~

(GREEDY) Sort the numbers in decreasing order and split the sequence in half. Now, construct the final sequence by taking alternate elements from the two halves. There will be 4 possible ways of doing this. Consider all of them.

Winner2 algo

~~~~~

Sort the list in increasing order . Let this be b_1, b_2, \dots, b_n Form two lists as follows. $b_1, b_4, b_2, b_5, b_3, b_6$ and $b_4, b_1, b_5, b_2, b_6, b_3$ for even n (here $n=6$) $b_1, b_5, b_2, b_6, b_3, b_7, b_4$ and $b_4, b_1, b_5, b_2, b_6, b_3, b_7$ for odd n (here $n=7$) If in both lists there are consecutive equals then solution not possible else print that list in which no equal consecutive are present.

Winner3 algo

~~~~~

Sort

Even  $2n$  nos if  $a(n)=a(2n)$  no

else pick  $a(n)$   $a(2n)$   $a(n-1)$   $a(2n-1)$  .....

Odd  $2n+1$  nos : if  $a(n)=a(n+1)$  no

else pick  $a(n)$   $a(2n+1)$   $a(n-1)$   $a(2n)$  ....

and place  $a(n+1)$  appropriately if it fits

(End beginning or centre)

Test cases

~~~~~

yes: 1 2 3

yes: 1 2 2 3

yes: 3 2 2 1

yes: 2 1 3 2

yes: 2 3 1 2

yes: 2 2 1 3

yes: 1 3 2 2

no : 1 2 2 2 3

yes: 3 2 2 2 3

yes: 1 2 2 2 1

yes: 3 3 3 6 6 6

yes: 6 4 6 4 6 4 6

yes: 1 2 2 2 2 2 4 5 6 7

no : 1 2 2 2 2 2 4 5 6

yes: 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

yes: 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

no : 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

no : 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

no : 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

yes: 3 3 3 3 3 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4

4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

4 4 4 4 4 4 4 4 3 3 3 3 3 4 4 4 4 3 3 3

3 3 3 3 3 4 4 4 4 3 3 3 3 3 3 3 3 3 3 3

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

3 3 3 3 4 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4

4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

4 4 4 4 4 4 4 4 3 3 4 4 4 4 3 3 3 3 3 3

3 4 4 4 3 3 3 3 3 3 3 3 3 3 4 4 3 3 3 3

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

~~~~~  
 |-----|  
 | Problem 2: Setsplitting |  
 |-----|

### The Problem

~~~~~

Bheeshma looked grimly at Krishna and said, `` You must decide, O Yadava, whom you will give your armies to. They cannot remain neutral in the great war that will take place.' ' `` But I love both the Pandavas and Kauravas. If I give, I give equally, or I do not give at all.' ' , replied Krishna, petulantly. Bhishma beamed. `` That is as easily done as winning a game of Snakes and Ladders with Shakuni at your side. Just give Yudhistira half your men, and Duryodhana the other half.' ' , he said.

`` Not so, Pitamaha! My army is comprised of many akshauhinis and the men of an akshauhini fight together. So, I have to give an entire akshauhini to either brother, and not parts of it. Unfortunately, each akshauhini has a different number of soldiers, and the math I learnt at my Gurukul is inadequate to help me figure out how to distribute the akshauhinis. What do I do?' ' , wailed Krishna. Bheeshma looked on, nonplussed. This was going to be a tougher nut to crack than that fisherman chap so many years ago....

Write a program that given the number of soldiers in each of Krishna' s akshauhinis determines whether he will send his soldiers to war or not. The input consists of $n+1$ lines, for n akshauhinis, the first line containing a single integer which is the number of akshauhinis and every other line containing a single integer which is the number of soldiers in the corresponding akshauhini. The output should be the word `` no' ' on a line, if there is no equitable distribution possible. If there is, then output the distribution in the following manner: for each akshauhini that goes to the Pandavas, print a +, followed by a space, and the number of soldiers in that akshauhini. For the Kauravas, substitute the + with a -. Each number (with its appropriate sign) should appear on a line by itself. There are at most 100 akshauhinis, with 0 to 100 soldiers in each, inclusive. ... 20 marks

In Short

~~~~~

You' re given a sequence of numbers, you' re supposed to insert + ' s and - ' s in between so that it evaluates to zero. i.e. find a subset of sum  $S/2$ , where  $S$  is the sum of all numbers (assumed positive). The numbers are given to be  $N$  in number and bounded by  $M$ .

### For example

~~~~~

1 2 3 4 : + 1 - 2 - 3 + 4 or - 1 + 2 + 3 - 4
 1 2 3 10 : no

My approach $O(MN^2)$

~~~~~

Have an array of size  $M*N$  which marks all the sums that can be made

## Winner1 algo

(BRANCH AND BOUND) Recursively enumerate all the subsets of  $\{1, \dots, N\}$ , where  $N = \text{\#inputs}$ . Whenever the running sum of a subset exceeds half the total sum, abandon that path.

We used Brute force. Let the list be  $a_1, a_2, \dots, a_i, \dots, a_n$ .  
Initially  $i=1$  and  $av(1)=(a_1+a_2+\dots+a_n)/2$ .  
Prob: pick some numbers in  $a_i$  to  $a_n$  such that total is equal to  $av(i)$ .  
step: If we can include  $a_i$  solve the prob with  $av(i+1)=av(i)-a_i$   
and  $i=i+1$ . If we can't include  $a_i$  solve prob with  $av(i+1)=av(i)$   
and  $i=i+1$ .  
If  $i>end$  : Solution not possible for that combination.  
If at any stage  $num(i)$  is -ve : Sol not possible for that combination.  
If at any stage  $num(i)=0$  : Sol found.  
If solution is not found for  $a_1$  then solution not possible.

Greedy:  
Sort  
Pick the largest as long as feasible  
Feasible when fits and required no. > least no.

~~~~~

----- | Problem 3: Areas | -----

The Problem

~~~~~

Star Date 6876.21. The known Universe is under the control of Earth. To facilitate easy patrolling, all of Earth' s dominion has been mapped onto a two-dimensional grid of squares. The grid is described by equidistant points, with each pair of adjacent points being joined by lines. Each square in the grid is a colony of Earth, and any spaceship must travel only along the lines of the grid.

Recently, a renegade band of Klatchians have been making furtive raids on our colonies. A Klatchian ship materialises at some point on the grid and drops its crew there. The crew travels from point to point in some random order, and after some time returns to the point where they started so that the ship can pick them up again. During this journey, they may revisit points and travel lines they have already gone through. All the colonies circumscribed by the travel line of the Klatchians are laid to waste. Earth Patrol has managed to capture an abandoned ship used in such a raid that contains only a log mentioning the details of the crew' s trip along the grid lines. Each line of the log contains a letter, ( n, s, w, e), which gives the direction of travel (north, south, west, east) and a number that denotes the number of lines travelled in that direction. For example, the log, ( s 1 w 1 n 1 e 1 ) is a trip that destroys exactly one square, while ( s 2 w 2 n 2 e 2 ) destroys 4. A longer sequence, with a diagram describing the travel is attached at the end.

Write a program that given the details of a Klatchian trip finds out how many colonies have been destroyed. The input is as specified above in the example, and the output should be an integer with no spaces before or after it, and succeeded by a newline, that gives the number of colonies. There are at most 200 lines of input, with the number on any line not exceeding 200. ... 25 marks

#### In Short

~~~~~

A particle moves on a grid in an arbitrary manner, but finishing at the starting point. What' s the area of the boundary of the region it encloses?

My approach

~~~~~

Find a point outside and floodfill, by doing a DFS on the outside region.

#### Winner1 algo

~~~~~

($O(M^3)$ ALGO) The alien travels along some M lines, say. Extend all these lines in both directions. This gives rise to some grid of rectangles. For every rectangle in this grid, find out if it is bounded on all 4 sides by line segments from the alien' s path (standard test for point inclusion in a convex figure). If yes, add this rectangle' s area to the current area.

Others have not submitted algos

~~~~~

### Test Cases

~~~~~

1)

4, e 1, n 1, w 1, s 1 . -- .

Area = 1

2)

4, n 1, w 1, e 1, s 1 . -- .

Area = 0

3)

```

. -- .
|   |
. -- .
|   |
. -- .
|   |
. -- .
|   |
. -- .

```

4)

```

. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .

```

5)

```

. -- .
|   |
. -- .
|   |
. -- .
|   |
. -- .

```

6)

```

. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .

```

7)

```

. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .
|   |   |
. -- . -- .

```

8)

Basic pattern:

9)

10) over and over the same square
still area = 1

The Problem

A scientist working the swampy, mosquito-infested jungles of

Sumatra has made an interesting find. She has recovered fragments of the gene code of a long-extinct animal, the Blarky, each fragment consisting of a string of molecules. The Blarky's genetic code is much like ours, with each molecule represented by a letter, except that all 26 letters are used, unlike our code that uses only the four letters, A, T, C and G to code the molecules in the gene.

Unfortunately, no single sample is guaranteed to be the entire genetic code of the Blarky, though it is known that every sample contains a string of consecutive molecules in the actual code. To reconstruct the final genetic code, our scientist relies on a well-known principle in genetics that states that the best approximation of a genetic code from its fragments is obtained by finding the shortest sequence of molecules that contains all the known fragments. For example, if the fragments gyy, segyy and puse are available, then the best guess at the code is pusegyy.

Write a program that, given gene fragments, obtains the best approximation. The input consists of $n+1$ lines when there are n gene fragments. The first line has a single integer, which is the number of fragments. All other lines have a single sequence of letters, there being no space between letters. The sequence on each line represents a different gene fragment obtained by the scientist. The output should be the genetic code as guessed from these fragments. Each fragment is at most 20 molecules long, and there are at most 100 fragments with the scientist. ... 38 marks

In short

~~~~~

You are given some strings, and you are expected to find the shortest string that contains all the other given strings, i.e. the shortest superstring.

My approach

~~~~~

NP-hard, and can be mapped to finding the Hamiltonian path. Hence approximate solutions (within 91% of my exact ones) are given full credit.

Winner1 algo

~~~~~

(GREEDY) At any stage, find the pair of strings giving the maximum overlap on merging in the best possible manner. Merge them, and proceed. At the end, the string left is the approximate answer.

Winner2 algo

~~~~~

Sort the strings in descending order of length. Pick the longest and copy it in the final string. Take the next one. Insert it as follows.
 If the final string is "abcdef" and the string to be inserted is
 1. "pqr". Final becomes "abcdefpqr".
 2. "def". Final remains the same.
 3. "defd". Final becomes "abcdefd"
 4. "fabcd". Form two strings "fabcddef" and "abcdefabc" and Final becomes the shorter of two.
 Keep on inserting strings in the descending order. Its a NP hard

problem. No polynomial time algo exists for it. Our algorithm works for fairly large number of cases.

Winner3 algo

~~~~~

Just removed strings that were already present

Test cases

~~~~~

1. all overlaps are zot
: any algo will work
2. aaaaa aaabbb aaabbb bbbccc
: two strings identical
3. aaaaa aaaab aabac baaab aabaab
: general arbit
4. cdefghi abcdef hijklm efghij
: two possibilities exist, which do you choose ?
5. raman mantra uma
: greedy does not work:
greedy takes raman+mantra = ramantra, and then uma: ramantrauma
you should take mantra+raman = mantraman, and then umantraman
6. abcd abef cdab efab
7. hello hell llama lotus shell ellot
: one string is a substring of another
8. finger oafing german muffin codger
9. life feels silli
10. with all the might of the earth rama lifted his bow and in just
one stroke fired twenty types of arrows which made all around him
stop to think about his powers
: to check exhaustive searches
11. who holds the hell torch is none other than the guy who lives here
or there and doth not know fear
12. abcdef defab fababc babc labcd gabcde gfab bcdfab ababef cdlab gbcd bcdg
13. csea prog contest is the worst thing that took place ever
14. sc re w t he gu y w ho do es an ex ha us ti ve se ar ch so th at he
lo ok s fo r be tt er so lu ti on s
15. abab abbc bbcad dacca aabdba abdbabd abadbabdbdbababab adbdbdababab abdbab
adbabad adbabad adbadb adbadb adbbdaba dabalbad dbadbabadabb dbadab dba
: general arbit