



Indian Association for Research in Computing Science

Excellence in Computing. Provably.

[HOME](#)[ACTIVITIES](#)[OLYMPIAD](#)[MEMBERS](#)[ABOUT](#)

Zonal Informatics Olympiad 2002, Part B

Question Paper

[IARCS home](#) > [OLYMPIAD](#)

Instructions:

1. The test consists of two problems. You have three hours to work out and submit the solutions.
2. Solutions must be in Pascal, C or C++. Use only standard features of language. For evaluation, you will submit source code that we will compile to test for correctness.
3. Save the source code of your solution to the two problems in separate files. When you submit your solutions, you will be asked to supply the file names of the source files.

Questions:

1. Write a program that reads an integer n as input and prints out the decimal representation of $1/n$. If $1/n$ is a recurring decimal, print out *two* copies of the recurring portion. You may assume that n is between 1 and 10000.

Here are some sample inputs and outputs for this problem:

- *Input: 2*
Output: 0.50 or 0.5
- *Input: 3*
Output: 0.33
- *Input: 5*
Output: 0.20 or 0.2
- *Input: 7*
Output: 0.142857142857
- *Input: 11*
Output: 0.0909

Related:

Overview
Current
Study Material
Online Judge
Archives

2. Write a program that reads the following sequence of inputs:

- An integer n between 1 and 10000
- n values of type real/float, to be stored in an array a with index values in the range $[0..n-1]$
- n integer values, to be stored in an array p

The n integer values stored in p represent a rearrangement (permutation) of the integers from 0 to $n-1$. In other words, p contains each integer from 0 to $n-1$ exactly once, but not in any particular order.

The problem is to rearrange the elements of a according to the rearrangement specified in p . For instance if $n = 3$, $a = [2.1, 3.5, 7.6]$ and $p = [2, 1, 0]$, then a should be rearranged as $[7.6, 3.5, 2.1]$. In other words, $a[0]$ is replaced by $a[2]$ because $p[0]=2$, $a[1]$ remains $a[1]$ because $p[1]=1$ and $a[2]$ is replaced by $a[0]$ because $p[2]=0$.

After rearranging the elements in a according to p , your program should print out the contents of a from $a[0]$ to $a[n-1]$.

Important restriction: Your program should not use any arrays other than a and p .

Note: You *must* store the rearranged values in a . It is not sufficient to just print out the rearrangement. The output should be printed using a simple loop that runs through the rearranged version of a from position 0 to position $n-1$.

Here are some sample inputs and outputs for this problem:

- *Input:*

3

2.0

3.5

7.0

2

0

1

Output

7.0, 2.0, 3.5

- *Input:*

5

2.0
3.5
7.0
10.5
13.0

2
4
3
1
0

Output

7.0, 13.0, 10.5, 3.5, 2.0