Due on 2020-03-18, 23:59 IST

Each morning, a bureaucrat receives letters, each letter has an identifying number, and denotes a job that takes one hour's work. The bureaucrat places the letters in an "In-tray", one above the other, with the letters received earlier below those received later. Each day the bureaucrat works for at most 8 hours. In each hour, he picks up the letter if any from the top of In-tray, does the job and throws away the letter. If the jobs finish early, the bureaucrat goes home early; if the jobs do not finish in the 8 hours, the letters remain in the In-tray. The next day, the arriving letters are put on top of the In-tray, as usual.

Write a program to model this process. Its input should be in the following form.

First there is a number n, denoting the number of days to be modelled. Following that there are n sequences of numbers, the ith sequence describing the letters arriving on the ith day. Each sequence starts of with a number giving the number of letters in the sequence, following which are the ids of the letters (in arrival order).

Your program is to print the ids of the letters left unprocessed at the end of the n days, if any, from the bottom of the In-tray to the top. You should assume that the In-tray will never have more than 20 letters.

Input instance (it is OK if the letter ids repeat)

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>11</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

Output

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>

Sample Test Cases

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Test Case 1

3
8 1 2 3 4 5 6 7 8
4 11 12 13 14
0

Test Case 2

4
1 10
2 1 2
0
9 1 2 3 4 5 6 7 8 9

Test Case 3

2
10 1 2 3 4 5 6 7 8 9 10
9 11 2 3 4 5 6 7 8 9

The due date for submitting this assignment has passed.
As per our records you have not submitted this assignment.

Sample solutions (Provided by instructor)

```cpp
#include <iostream>
#define repeat(x) for(int _iterator_i = 0, _iterator_limit = x; _iterator_i < _iterator_limit; _iterator_i++)
#define main_program int main()
#include <cmath>
using namespace std;

int main()
{
    int intray[25], top = 0;
    int n; cin >> n;
    for(int i=0; i<n; i++)
    {
        int nToday; cin >> nToday;
        for(int j=0; j<nToday; j++)
        {
            int letter; cin >> letter;
            intray[top] = letter;
            top++;
        }
        top = max(top-8,0);
    }
    for(int i=0; i<top; i++)
    cout << intray[i] << endl;
}
```
Week 8

Lecture 16:
Array Part-2:
Part 1: Introduction
(unit? unit=92&lesson=107)

Lecture 16:
Array Part-2:
Part 2: Interpretation of aname[index]
(unit? unit=92&lesson=108)

Lecture 16:
Array Part-2:
Part 3: Arrays and function calls (unit? unit=92&lesson=109)

Lecture 16:
Array Part-2:
Part 4: A function to sort an array (unit? unit=92&lesson=110)

Download Videos (unit? unit=92&lesson=183)

Weekly Feedback (unit? unit=92&lesson=195)

Quiz: Week 7 Quiz (assessment? name=203)

Week 7 Programming Assignment 1 (/noc20_cs53/progassignment? name=204)

Week 7 Programming Assignment 2 (/noc20_cs53/progassignment? name=205)

Week 7 Programming Assignment 3 (/noc20_cs53/progassignment? name=206)

https://onlinecourses.nptel.ac.in/noc20_cs53/progassignment?name=206
<table>
<thead>
<tr>
<th>Text Transcripts</th>
</tr>
</thead>
</table>

Week 9

Week 10

Week 11

Week 12