Grading Students


by nabila_ahmed

HackerLand University has the following grading policy:

- Every student receives a $grade$ in the inclusive range from 0 to 100.
- Any $grade$ less than 40 is a failing grade.

Sam is a professor at the university and likes to round each student's $grade$ according to these rules:

- If the difference between the $grade$ and the next multiple of 5 is less than 3, round $grade$ up to the next multiple of 5.
- If the value of $grade$ is less than 38, no rounding occurs as the result will still be a failing grade.

For example, $grade = 84$ will be rounded to 85 but $grade = 29$ will not be rounded because the rounding would result in a number that is less than 40.

Given the initial value of $grade$ for each of Sam's n students, write code to automate the rounding process. For each $grade_i$, round it according to the rules above and print the result on a new line.

Input Format

The first line contains a single integer denoting n (the number of students).
Each line i of the n subsequent lines contains a single integer, $grade_i$, denoting student i's grade.

Constraints

- $1 \leq n \leq 60$
- $0 \leq grade_i \leq 100$

Output Format

For each $grade_i$ of the n grades, print the rounded grade on a new line.

Sample Input 0

```
4
73
67
38
33
```

Sample Output 0

```
75
67
40
33
```

Explanation 0

ID	Original Grade	Final Grade
1	73	75
2	67	67
3	38	40
4	33	33

1. Student 1 received a 73, and the next multiple of 5 from 73 is 75. Since $75 - 73 < 3$, the student's grade is rounded to 75.
2. Student 2 received a 67, and the next multiple of 5 from 67 is 70. Since $70 - 67 = 3$, the grade will not be modified and the student's final grade is 67.
3. Student 3 received a 38, and the next multiple of 5 from 38 is 40. Since $40 - 38 < 3$, the student's grade will be rounded to 40.
4. Student 4 received a grade below 38, so the grade will not be modified and the student's final grade is 33.

Submissions:91590

Max Score:10

Difficulty: Easy

Rate This Challenge:

More

Current Buffer (saved locally, editable)

Java 7

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    static int[] solve(int[] grades, int n){
        // Complete this function
        for(int i = 0; i < n; i++){
            if(grades[i] >= 38){
                if((grades[i] - 3) % 5 == 0){
                    grades[i] += 2;
                }else if((grades[i] - 4) % 5 == 0){
                    grades[i] += 1;
                }
            }
        }
        return grades;
    }

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int n = in.nextInt();
        int[] grades = new int[n];
        for(int grades_i=0; grades_i < n; grades_i++){
            grades[grades_i] = in.nextInt();
        }
        int[] result = solve(grades, n);
        for (int i = 0; i < result.length; i++) {
            System.out.print(result[i] + (i != result.length - 1 ? "\n" : ""));
        }
        System.out.println("");


    }
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code