[CS 11 25.1] Lab 6e – Finals Week

Cheatsheet is available here: https://oj.dcs.upd.edu.ph/cs11cheatsheet/

Submit solution [CS 11 25.1] Lab Exercise 6

My submissions

Points: 230 (partial)
Time limit: 6.0s
Memory limit: 2G

Problem type

Allowed languages
py3

Problem Statement

It's finals week! Guds Tudent is preparing for their final exams, so for each of the next n days, they will prioritize exactly one of the following tasks: (i) studying, (ii) doing chores, or (iii) hanging out with friends.

On the i^{th} day, prioritizing studying, chores, and friends will give Guds a_i, b_i, and c_i happiness, respectively. As this is finals week, Guds cannot hang out with friends on two consecutive days.

What is the maximum amount of happiness Guds can attain after n days?

Task Details

Your task is to implement a function named `max_happiness`, which should have the following *signature*:

```
def max_happiness(a, b, c):
```

Copy

It has three arguments a, b, and c. These are length-n sequences of integers (`int`s) as described in the **Problem Statement**.

The function must return an integer denoting the maximum happiness Guds can attain after n days.

Restrictions

- The following symbols are allowed: `iter`, `next`, `map`, `filter`
- The following imports are allowed:
  - `count`, `islice`, `chain`, `takewhile`, `starmap` and `zip_longest` from `itertools`.
  - `cache`, `lru_cache`, `total_ordering`, `partial`, `reduce` and `wraps` from `functools`.
  - `randint`, `randrange and choice` from `random`.
  - `Fraction` from `fractions`.
  - `dataclass` from `dataclasses`.
  - `contextmanager` from `contextlib`.
  - `Enum`, `auto` from `enum`.
- Anonymous functions are allowed.
- Inner functions are allowed.
- Classes, dataclasses, and enums are allowed.
- Recursion is allowed.
- Loops are allowed.
- Generators and comprehensions are allowed.
- Your source code must have at most 1000 bytes.

Examples

Example 1 Function Call

```
max_happiness([1, 1, 1, 11], [10, 10, 10, 11], [100, 100, 100,
11])
```

Copy

Example 1 Return Value

```
221
```

Copy

Constraints

- The function `max_happiness` will be called at most $70{,}000$ times.
- $1 \leq n \leq 350{,}000$
- $|a_i|, |b_i|, |c_i| \leq 10^{10}$
- The sum of n across all calls to `max_happiness` will be $\leq 350{,}000$.

Scoring

Note: New tests may be added and all submissions may be rejudged at a later time. (All future tests will satisfy the constraints.)

- You get 70 ❤️ points if you solve all test cases where:
  - $n \leq 10$
  - The sum of n across all calls to `max_happiness` will be ≤ 100.
- You get 50 🔴 points if you solve all test cases where:
  - $n \leq 50$
  - The sum of n across all calls to `max_happiness` will be ≤ 800.
- You get 10 ❤️ points if you solve all test cases where:
  - $n \leq 400$
  - The sum of n across all calls to `max_happiness` will be ≤ 800.
- You get 10 ❤️ points if you solve all test cases where:
  - $n \leq 6{,}000$
  - The sum of n across all calls to `max_happiness` will be $\leq 12{,}000$.
- You get 90 🔴 points if you solve all test cases.

Clarifications

Report an issue

No clarifications have been made at this time.