D.E.S.P.A.I.R.

Diogn's Eldritch Stories in Preparation for Annoying, Insightful Regrets

A **Mock HOPE Template** made with Love by **Diogn Mortera** (@mortrpestl)

What's the Point?

Mock HOPE lol /j I

I made these questions not primarily for you to code them but for you to exercise your insights and thought-making process.

Solutions and (most of the time) code for non-heartbreaks can be found in the end of the document and the folder where you probably found this document, respectively. If you can't solve it, read the solutions for insights 😄

On Grading Yourself

You can submit code in my *Offline Judge* 😎.
• Your code for most problems and their subtasks can be checked by going to the "(submit code here!) offline_judge" folder;
• going to the folder with the problem number/function name;
• typing your code into student_solution.py, and;
• running the file.

Verdicts will be provided as the file is run. Isn't that cool!

Do note that *there may be mistakes in the logic of my solutions/test cases I am unaware of*. Please let me know, but please do understand if ever 💔 there will be things that will slip under my proofreading 😌

• That said, I update the file and test cases regularly if I find them / someone points them out so regularly *redownload* the folder. Next MOCK Hope, maybe I have set up my own Online Judge though so that isn't a problem, but for now, please do that if you want updated versions of test cases 😄

Other Information

• *The difficulty can **fluctuate***. **SKIP** PROBLEMS *YOU THINK YOU CAN'T SOLVE/ WILL TAKE TIME* and TRY THEM IF AND ONLY IF YOU'RE SURE YOU'VE SOLVED ALL THAT YOU COMFORTABLY CAN.
  - This should be the general strat even in the actual HOPE. Some problems may be easier or harder depending on what you're good at. I tried my best to have a variety of problem types to this set.

• Knowing and (coding) the first **1-2** subtasks of each problem should be your goal. If you can do that, you're pretty well-off in HOPE.

  ▸ Some subtasks are ***special forms of heartbreaks***. They will be signified with a *reaction image* related to the question. When you see this, the heartbreak is ***definitely not easy*** to solve and probably not necessary for the HOPE either (but you never know). They will also usually not have model solutions because nobody is reading all of that.

• And of course, Don't stress yourself out.

Before you proceed: https://oj.dcs.upd.edu.ph/cs11cheatsheet/

Recall
(if you don't know/recall a term, try to learn it! or skip it…)

You May Include a Summary of Tricks/Important Things to Remember Here
• Like this…
• Or like this!!

Maybe even have multiple sections?
• More!

The **DESPAIR** questions can be found in the next pages.

Feel free to time yourself for *3 hours*.

[total with heartbreaks] is perfect. Try to get at least [total without heartbreaks] in 3 hours if you want to pass.

If you want to test, I recommend checking your solution against the examples for now, then checking it with the model solutions later (to avoid wasting time).

Good luck!

Problems

(1)**Sample Problem Here**(1)

To change this image, go to the "img" parameter of the problem container and change the filename. You may also change the width.

Use your freedom of expression here.

A problem should be as concise as possible, while still providing succinct and adequate explanation to the answerer.

What is the meaning of life? Given a and b, the meaning is $a+b$.

Task Details
Define `solve(a,b)`. You are free to add other parameters inside `solve()`, but it cannot have no parameters (as of October 2025).

Avoid changing the function name, or you will have to change more files in `tester_packet`. More on checking in the `tester_packet` README

The output should be a single `int`.

Examples
`solve(1,2)` results in 3

Constraints
If there are parameters, be absolutely clear in specifying them. Make sure your constraints match your test cases, especially if you have many subtasks.

Example:
$0 \le a, b \le 20$

💗 200 points: Insert easy subtask
🤩 169 points: No further constraints
💔 0 points: Insert buff here

(2)**Empty Problem for Template**(2)

Task Details

Examples

Constraints