

COGNIZANCE TASKS

Difficulty - Easy to Medium

This document comprises a set of tasks that you can try to complete to join Cognizance. These tasks are meant to help you in understanding the basics of various fields in CS; Moreover, they will help you in building confidence in reading and writing code. In the present world, learning to learn is one of the key skills that one should develop which will help you grow so approach these problems with the curiosity to learn and try to build your skills.

Submission Guidelines

1. Make a repository in GitHub with the name `cognizance-tasks`
2. Store the results and code in the respective folders each with the name `task-#` (where # is the task number).
3. Include a brief explanation of how you approached each task in a `README.md` file.
4. Make a folder named 'introspection'. In the folder, write answers to the following question in a `README.md`:
 - a. Why must we choose you?
 - b. What is your plan for the next 6 months?
 - c. How would you contribute back to the club?
5. On completion, let us know on Discord :)

Make sure you don't get sidetracked into learning a whole language for completing a task.

You only have to learn the necessary things which are required to complete the task. Google will be your best friend throughout the tasks, it [will be tricky](#) at times but [learning to Google efficiently](#) will help you greatly.

If you are a beginner with no prior experience with programming, Task-10 would be the right one, to begin with.

Please ask your questions in the discord channel whose link was sent along with the mail.

Note - Check your email inbox regularly so that you do not miss important communication from us.

0. Install Ubuntu 20.04 [Desktop] or any other comfortable Linux Distro

You can download it from the [Ubuntu website](#). You can also mention the errors that you encountered and the solutions which worked, in the discord server.

You may encounter errors while installing Ubuntu, so keep a backup of your data. If you already have Windows installed, you could dual-boot Ubuntu alongside Windows. For macOS users, you need not install Ubuntu but start using the terminal more ;)

If you need help in installing the same, [send us an email](#) or contact the seniors.

1. Programming

Solve the problems **Python/C/C++** or any other programming language. We will be evaluating everyone's code submissions, so please don't copy the code. **Plagiarism is not the way to get into the club :)**

Show us what you did and we'll be happy with it.

An elephant decided to visit his friend. It turned out that the elephant's house is located at point 0 and his friend's house is located at point x ($x > 0$) of the coordinate line. In one step the elephant can move 1, 2, 3, 4 or 5 positions forward. Determine, what is the minimum number of steps he needs to make in order to get to his friend's house.

Input

The first line of the input contains an integer x ($1 \leq x \leq 1\,000\,000$) — The coordinate of the friend's house.

Output

Print the minimum number of steps that the elephant needs to make to get from point 0 to point x .

Add your code in the `task-1` folder in the `cognizance-tasks` repository. You can add output screenshots to the same folder.

2. Debugging Derbies

Lenna was just trying to make a quiz application for maths using Java. But she seems to be facing a lot of issues in the code that she wrote. Could you go through the code and find out all the mistakes and fix them for her?

Once you manage to get the application running, test out each feature because Lenna isn't great at maths and would have made a lot of mistakes. One thing Lenna is good at is commenting on her code, so you shouldn't have an issue navigating through it.

For example, when you press on the addition button to add numbers, the calculator will actually multiply them. Hunt for all such bugs, and fix them. Bugs may range from minor typos in the calculator to, syntax errors, and then logical errors which make the calculator give a wrong answer.

You can get the code by running the following command in your terminal:

```
$ wget -O MathQuiz.java http://tiny.cc/cognizance-task2
```

After fixing all the bugs you encounter, upload the `MathsQuiz.java` file to the `task-2` folder in your `cognizance-tasks` repository.

References:

<https://en.wikipedia.org/wiki/Lenna>

<https://www.digitalocean.com/community/tutorials/how-to-install-java-with-apt-on-ubuntu-18-04>

<https://askubuntu.com/questions/145748/how-to-compile-a-java-file-on-ubuntu>

3. Rustic Scraper

Can you write a Fe_2O_3 (Rust) script that scrapes (web-scraping) information about the 10 most COVID affected countries with the details (total cases, total deaths, and total recovered) from [worldometers.info](https://www.worldometers.info) and write it to a CSV file?

Format the result, manually, as a markdown table in `README.md` in the `task-3` folder in your `cognizance-tasks` repository, and also add the script & the CSV file.

References:

<https://codeburst.io/web-scraping-in-rust-881b534a60f7>

<https://medium.com/@joydeepubuntu/rust-and-webscraping-ebecc9ae536c>

<https://guides.github.com/features/mastering-markdown/>

4. Geddit

Have you ever explored Reddit? People say it's a fun place to hang out, the only place which gives you good karma. The only way we want to go to Reddit is with "Go" - geddit?

Use golang to write a script that can:

- Perform a search for subreddits using 'memes' as the search query
- Pick the subreddit that occurs as the first search result
- Using that, upvote posts posted in the last week (limited to 100 posts at max)

Add the golang script to the `task-4` folder in your `cognizance-tasks` repository.

References:

<https://github.com/vartanbeno/go-reddit>

[Tutorial: Get started with Go](#)

5. Ubiquitous UI

After seeing an application run on a mobile phone, ever thought maybe someday you would like to make your own? How would you feel if we told you that you could make an application run on both Android and iOS mobiles by writing code in just one language? Well, today you get to do just that. Using Google's framework, Flutter, you can do this easily. For your task, you need to make a simple user interface just like the one given below.

Sample UI: <https://github.com/cognizance-amrita/tasks/tree/main/task5>

Add your project files into the `task-5` folder in your `cognizance-tasks` repository. Add a `README.md` to the folder as well and add a short screen recording of the UI you made.

References:

[Flutter Installation](#) | [Flutter Docs](#)

[Tabs in Flutter](#) | [Making the dot animated when changing tabs](#)

[Image assets to be used as shown in the demo](#)

Note: To be able to run the same application on an iOS device, you need to have a Mac with XCode in it. You can run the application on just an Android/iOS/emulator device as per the hardware available to you.

6. Sir Perceval's quest

Send Sir [Perceval](#) on a quest to retrieve all the commits from [Cognizance](#) organization projects.

You can use the Perceval tool to retrieve and gather data from git repositories. You can collect the commits' information from the `cognizance-amrita/cognizance_cms` repository using the following commands.

```
$ perceval git --json-line  
https://github.com/cognizance-amrita/cognizance_cms >> commits.json  
$ cat commits.json
```

The task is to write a script fetching the repository names (using GitHub API) of an organization (say `cognizance-amrita`) and fetch the commits of all those repositories using Perceval (given example is for one repo) and dump them in the `commits.json` file. Please add the script as well as the `commits.json` file in the `task-6` folder in the `cognizance-tasks` repository.

[Repositories | GitHub Docs](#)

[Automate the Boring Stuff with Python \(Ch 12, 16\)](#)

7. Spammer_Spaghetti

Do all modern web browsers have a console in them? What can you do in this console? Spam your lazy friend with lots of 'Hi' messages on the telegram web using JavaScript.

Please add the JavaScript file in the `task-7` folder in your `cognizance-tasks` repository.

https://www.w3schools.com/js/js_htmlDOM_html.asp

8. CS50

Watch all [CS50 lectures](#) and complete problem sets till week-4. Save the work that you have done and try to work on the [CS50 IDE](#) itself. Please update your status in the GitHub repository and have your problem solutions in a separate `pset` sub-folder.

Additional Tasks

9. Poster Design (Bonus points if attempted)

Pictures convey a million words, try expressing yourself with a poster that describes yourself.

Design a poster that best describes yourself. You can use canva.com/Adobe Photoshop or similar tools.

Upload the poster in the respective folder.

10. If maths was fun :P

<https://www.codechef.com/ICM2020/problems/ICM2003>

11. Circuit Design (for Hardware freaks)

The task is to design and code a circuit (Arduino) to perform the following task.

Your circuit should be able to stop the motion of a motor when your circuit detects an obstacle within 100cm. And use a mechanism of your choice to notify the user of the obstacle

To simulate use <https://www.tinkercad.com>

<https://www.arduino.cc/en/Guide/HomePage>

<https://www.instructables.com/id/Arduino-Connections/>

The code should be written strictly with no drag and drop code.

Upload the solution and screenshot of the circuit to the respective folder.

12. Bandit

Complete Bandit till level 10.

Refer: <http://overthewire.org/wargames/bandit/>

Please store the password of each level in a `pass.txt` file and upload it in the respective folder.

13. Project Euler (for math geeks)

<https://www.hackerrank.com/contests/projecteuler/challenges/euler001>

<https://www.hackerrank.com/contests/projecteuler/challenges/euler002>

<https://www.hackerrank.com/contests/projecteuler/challenges/euler003>

<https://www.hackerrank.com/contests/projecteuler/challenges/euler004>

<https://www.hackerrank.com/contests/projecteuler/challenges/euler005>

Upload all the solutions to the respective folder.