

# Programming Competition

## Kryptik 2024

Welcome to the Programming Competition for Kryptik 2024! As you embark on this journey of problem solving and learning, this guide will help you set up your development environment, understand the competition structure, and adhere to the code of conduct.

## Setting Up Your Environment

Before you dive into the problems, it's essential to set up your development environment to ensure that your code runs in the same conditions as everyone else's. Below are the step-by-step guidelines to get you started.

- 1. Install Miniconda or Anaconda:** If you haven't installed Conda on your system, you'll need to do so. Depending on your preference, you can choose either:
  - **Miniconda** (<https://docs.conda.io/projects/miniconda/en/latest/miniconda-install.html>). (Lightweight, only necessary packages).
  - **Anaconda** (<https://docs.anaconda.com/free/anaconda/install/index.html>). (Comprehensive, includes many Python tools, libraries, a CLI and more).
- 2. Obtain Competition Files:** Ensure you have all necessary files, including `kryptik_2024_env.yml`, as it contains the required environment configuration.
- 3. Create the Conda Environment:** Navigate to the directory containing `kryptik_2024_env.yml` and run:

```
conda env create -f `kryptik_2024_env.yml`
```

**Note** It can take some time to generate the environment.

- 4. Activate the Environment:** Every time you work on the competition, ensure you activate the environment using:

```
conda activate kryptik_2024_env
```

## Working on Problems

With the environment activated, you can now work on the competition problems.

- **Testing Your Solution:** Use `pytest` for evaluating your solutions. Example:

```
pytest test_filename.py
```

For instance, to test your solution for the first problem, you would execute:

```
pytest test_part_1.py
```

**Note:** Passing the provided tests is just the beginning. Our comprehensive evaluation tests cover additional scenarios, so consider edge cases in your solution.

## File Structure and Overview

The `CRC_2024_KRYPTIK_PRELIM_1` directory has been structured to provide clarity and ease of navigation. Here's a breakdown:

- **README.md:** This very document! Contains instructions, general information about the competition, and guidelines on using the provided files.
- **Prelim1\_FR.pdf:** Contains the problem questions in French.
- **Prelim1\_EN.pdf:** Contains the problem questions in English.
- **kryptik\_2024\_env.yml:** The Conda environment file. This file contains the specifications for the Python environment you should use to ensure code consistency and compatibility.
- **Prelim1/:** This directory holds all the individual problem sets for the competition.
  - **part\_1** (and similar directories for subsequent problems):
    - **part\_1.py:** A Python template file where you will write or complete the code for the problem. This template might have some boilerplate or helper code to get you started.
    - **test\_part\_1.py:** The abridged test suite for `part_1.py`. You can use this to test your solution against a set of pre-defined cases. Take note that when we evaluate your solutions, our test suite will test more cases that are not covered in the given test suite.

- 

## Submission Guidelines

### Packaging Your Solution

When you've completed the problems and are ready to submit, please follow these steps:

1. **Integrity of Files:** Do not delete or rename any files from the `CRC_2024_KRYPTIK_PRELIM_1` directory.
2. **Packaging:** Zip the entire `CRC_2024_KRYPTIK_PRELIM_1` directory. Ensure the zip file maintains the directory's structure.

3. **Submission:** Send the zipped folder to the google drive link provided in your participant kit. If you don't have the link, ask your mentor for the infos from the participant kit and if needed contact CRC.Frank or CRC.Louis on discord.

Only one submission (zip file) will be accepted per school.

## Code of Conduct & Integrity

Maintaining integrity and ethical conduct is paramount.

- **Participant's Work:** All submitted solutions should be done only by the participants of the competition.
- **No Plagiarism:** Copying solutions, whether from other participants or from online sources, is strictly prohibited. Any participant found plagiarizing will be disqualified.
- **No Sharing of Solutions:** Do not share your solutions, discuss problem-solving techniques, or provide hints to others during the competition.
- **Use of Third-Party Libraries:** Unless explicitly mentioned, the use of third-party libraries, tools, or online services (other than for environment setup or documentation reference) is not allowed. Refer to the official rulebook (Section 4.3.1) for allowed libraries and if unsure ask in the CRC discord channel for programming questions.

Remember, the essence of this competition is to learn, challenge oneself, and enjoy the process of problem-solving. Breaching integrity not only affects your learning journey but also diminishes the efforts of other who are competing fairly.

## Frequently Asked Questions (FAQS)

### What if I encounter issues installing Miniconda or Anaconda?

First, ensure you're following the official installation guides for [Miniconda](https://docs.conda.io/projects/miniconda/en/latest/miniconda-install.html) (<https://docs.conda.io/projects/miniconda/en/latest/miniconda-install.html>) or [Anaconda](https://docs.anaconda.com/free/anaconda/install/index.html) (<https://docs.anaconda.com/free/anaconda/install/index.html>). If issues persist, your best friend is a good Googling session, where a lot of answers can be found. If nothings works, don't hesitate to ask us for help.

### Are there any restrictions on the programming languages we can use?

This year, for Kryptik 2024, the only programming language allowed is Python. More specifically, you will be using Python 3.10 as specified in the conda environment.

### Can we use external libraries or modules in our solution?

Unless explicitly mentioned in a problem statement, the use of third-party libraries and module is not permitted. The focus is on using Python's standard libraries to solve problem. If an external library is allowed, it will already be found in the provided conda environment. You can read the official rulebook (Section 4.3.1) for more information.

## What happens if we miss the submission deadline?

Submissions received after the deadline will not be eligible for judging. We encourage all teams to submit their work well before the deadline to avoid any last-minute issues.

## How will our code be evaluated?

Code submissions are evaluated based on correctness, efficiency, adherence to problem requirements, and code readability. We also use a comprehensive suite of tests, including the ones provided, to ensure robustness.

## Can we update our submission?

You can update your submission as many times as you like before the deadline. Only the latest submission will be considered for evaluation.

## What should we do if we find a mistake on the problem statement?

If you believe there is an error in the problem statement, please contact us immediately. We will review the query and provide clarification or corrections if necessary.

## How can we prepare for the competition?

Familiarize yourself with Python, computer science concepts and ensure you understand the use of `pytest` for testing your solutions.

## I don't know how to program in Python, where do I start?

Don't worry, the internet is filled with amazing resources to learn programming. One that we strongly recommend is from [freeCodeCamp](https://www.youtube.com/watch?v=eWRfhZUzrAc&list=PLWKjhJtqVAbnqBxcdjVGgT3uVR10bzTEB) (<https://www.youtube.com/watch?v=eWRfhZUzrAc&list=PLWKjhJtqVAbnqBxcdjVGgT3uVR10bzTEB>) on Youtube.