

National University of Singapore  
School of Computing  
CS1010S: Programming Methodology  
Semester II, 2022/2023

**Contest 2.2**  
**Beautiful Runes**

Release date: 26<sup>th</sup> January 2023

**Due: 5<sup>th</sup> February 2023, 23:59**

## Required Files

- contest02.2-template.py
- runes\_2D.py
- graphics.py
- PyGif.py

## Background

You have become adept as a PIM apprentice but so are many others like yourself. With everyone attempting to prove themselves superior, it is certain unhealthy rivalry will form amongst the fresh apprentices.

But the masters have already foreseen this problem through the many generations of PIM mages they have trained. Initially masquerading as a rumour, news of the annual rune conjuring contest quickly became the hottest of discussion topics.

With exquisite and intricate winning runes being displayed prominently in the grand hall and the hustle and bustle of preparation, you barely managed to get hold of a trainer to get the details. Clearly, it was not intended for all apprentices to participate but only those possessing true passion and are pure of essence. Do you have what it takes?

## Task:

This contest represents the 2D runes segment of the annual rune conjuring contest which you may participate in.

Being masters of rune manipulation, you are to use your creativity and design some cool-looking runes. Simply define your runes in the template functions provided.

You may submit up to three 2D runes. Submit your entries by including the code for each rune in the corresponding template function. If you are submitting less than three runes, leave the extra functions empty.

Just remember to submit and finalise your submission when you are done!

**Additional notes:** Please submit your runes in the order of Entry 1, 2, 3, if you are submitting less than 3 runes. Please do not submit duplicate runes. You are also not allowed to use external images in your entry.

**Warning!** The **runes\_2D.py** and **graphics.py** files are not to be edited. As such, you are **only** allowed to use runes that are already made available in the **runes\_2D.py** file. In particular, you are not allowed to define your own runes to accommodate for colour.

Since this is a 2D runes contest, you are **not allowed** to use any of the functions that are related to 3D-runes, such as `overlay`. These functions have been removed from the **runes\_2D.py** file for your convenience.

Finally, you should simply return the runes instead of showing the runes. Here's an example.

#### WRONG

```
def rune_entry_1():  
    your_rune = ...  
    return show(your_rune)
```

#### CORRECT

```
def rune_entry_1():  
    your_rune = ...  
    return your_rune # Return the rune instead of showing it!
```

### Viewing your rune:

To view your rune, individually uncomment the line that calls your rune function. For example, if you want to view your rune in Entry 1, uncomment the line

```
show(rune_entry_1())
```

**Warning!** Do not uncomment more than one rune at a time, if you do you might not see the rune you want to see. If you want to view your rune in Entry 2, uncomment the 'show' statement for Entry 2 and comment Entry 1.

### Viewing Your Rune as a file:

**[Optional] For advanced users only!** Often times if your rune is extra large, our graphics module will not be able to save an image file that matches what you see on your screen. In this case you might want to download your rune as a file to check what we will be uploading. To do this, simply uncomment the following line to the end of your rune function:

```
save_image('rune_entry_1') # Uncomment this to save your rune
```

You will have to delete the previous copy of the image if you want to save a new copy of the image again.