

Last time we worked with Trinket you were working on a simulation that took a list of numbers and added them all together.

After given some “behind the scene” instructions as to how the computer interoperates instructions, you completed the code that made the simulation work!

<https://trinket.io/glowscript/a5f49f8cf4>

The code in lines 6 through 9 was added by you and allowed the simulation to work!

Now you will do the same thing with a much more complicated program, and something that is a bit more physics related.

<https://trinket.io/glowscript/581c4f32d8>

This new simulation allows you to draw vectors on a grid, and then computes the sum of these vectors (the last simulation only allowed you to draw the vectors, no addition, but allowed you to do so on any map/image).

However, the code that is needed to add the vectors together is missing. Luckily you know how to create a for loop that will go through a list and add all the things on that list together!

You will add code on lines 72 through line 75 (four lines just like last time!)

There are a few important differences this time around. Since this is a much more complicated program, the list is created and named elsewhere in the code.

As such the name of the list you will go through is: arrows

You will create a for loop inside of the function `sum_of_vectors()` that will go through the list arrows, and add all of the vectors in that list together.

Since we are working with vectors and not simple numbers (scalars) you will need to initiate a variable as a vector.

To do so use the following example:

```
vector_variable = vector(a, b, c)
```

where `vector_variable` will be the new variable you create to represent the sum, and a, b, and c are the numbers that you initiate that vector with.

The last difference is that instead of using items from the list, and adding these items to the sum, we need to adjust things to suit the vector sum as oppose to normal addition.

To add things to your new vector that you have previously initiated you will need to attach “.axis” to what you are adding (this is probably the least well explained part so ask for help when needed!).

Last, just like last time, we need to “return” the vector that is all the other vectors added together.



This work is licensed under the Creative Commons Attribution 4.0 International License.
To view a copy of the license, visit <https://creativecommons.org/licenses/by/4.0/>.