

# Information about sample I/O files

**Release Date: 21/03/2024**

Because there were some mistakes in the first version of I/O files, we are giving you new files that has same format and style as the ones before (and as shown in your introduction pdf), but has different data. Please note that old sample I/O files ARE NOT VALID now. Please test your code with these new files instead.

Additionally since some of your friends wanted some more samples to get a better understanding of different scenarios, this version of I/O folder includes not one but three sample packs. Since all them include some different examples, try your program with all of them and make sure your code matches with the data on all of their output files. Bellow are some notes about these files.

## Sample I/O Pack Number 1

A simple example where machine is filled with no problems. Includes:

- "INFO: Insufficient money, try again with more money."
- "INFO: Number cannot be accepted. Please try again with another number."
- "INFO: Product not found, your money will be returned."
- "INFO: This slot is empty, your money will be returned."
- And purchases with no problem.

## Sample I/O Pack Number 2

A sample with big Product.txt input so you can see an example with full machine. Includes:

- "INFO: There is no available place to put {product name}"  
(This message is used when there are no available slots that fit the product we are trying to load, and the loading has not been stopped yet. For example, if last slot has Water(0, 3), so there is 7 more spaces in the slot, but we are trying to load a Chicken Wrap.)
- "INFO: The machine is full!"  
(Written when machine is completely full. As you will see in the example, this message is only printed once. After this message you should stop trying to load more products into the machine.)
- And some other examples like pack number 1.

## Sample I/O Pack Number 3

A sample with:

- An example where all the products in one slot are sold.  
(So you can see that a slot is not assigned to a particular product indefinitely. If products in a slot is sold out, the slot should be reset and look like all the other empty slots.)
- An example with 0 calorie input in a scenario where there are no products with 0 calorie.

(So you can see that while you are writing GMM to output file you are to write 0 for the calorie if the slot is empty, but this does not mean that you can point to an empty slot if there are no products with 0 calorie.)

- And some other examples like pack number 1.