

3. Factorio

So instead, I'm going to have you work on a very small part. The Centrifuge is a structure that takes 10 uranium ore and after 12 units of time (measured in game ticks) generates an expected 0.007 units of enriched U235 and 0.993 units of depleted U238 for each cycle of running. The use of these uranium isotopes is left to the player.

Details found here: https://wiki.factorio.com/Uranium_processing

You should write a function named "Centrifuge" in the file "Factorio/main.cpp". It takes three arguments: amount of time, amount of ore, amount of U238. It should modify its arguments after running the most cycles it can and should return the amount of U235 generated. Please assume expected amounts of U238 and U235 per cycle (no probability needed).

Example test case, "Too much time"

```
int time = 900;
int ore = 45;
double u_238 = 0;
double u_235 = Centrifuge(time, ore, &u_238);
CHECK(time == 852);
CHECK(ore == 5);
CHECK(std::abs(u_238 - 3.972) < 0.001);
CHECK(std::abs(u_235 - .028) < 0.001);
```

Explanation

With 900 ticks of time and 45 ore, the centrifuge can run for 4 cycles. This will leave 852 ticks left and 5 ore remaining. As there wasn't any U238 to start with so U238 is the 3.972 units that was created. The std::abs and subtraction is needed to ensure the floating point imprecision doesn't cause test case failure. (I'd rather say `u_238 == 3.972`, but that can be false due to floats math.) Lastly, .028 units of U235 are expected.



Argument types

Please note that I haven't specified the argument and return types. But you should be able to infer them by closely examining the test cases and the compiler errors.