Battery Take Home Question

Hiring an "Energy Software Developer" isn't really a thing. We are interested in good generalized software developers, and don't expect new employees to have background knowledge on the power industry. To give you an idea on what we work on at Wärtsilä, we have a coding challenge that should only need basic addition, subtraction, multiplication, and division to complete.

We'd like you to calculate the state-of-charge of a battery as it goes through charging cycles, and write some code to demonstrate your ability to express the algorithm in a readable manner to other developers.

Background

When a customer installs a large battery in their power plant, one of their main functions is to store energy and release it at a later time. Customers like to save money, and being able to "shift" energy around helps them manage the reliability of the electric grid to which they are connected, and in the process, earn money from the electric utility.

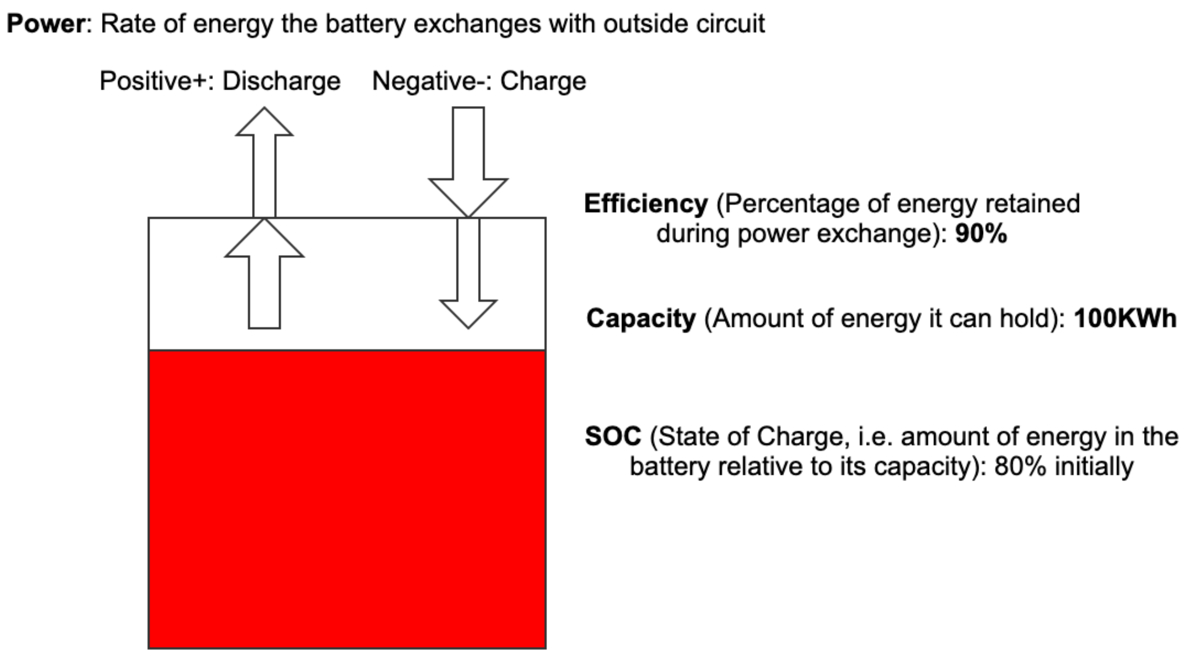
Unfortunately, physics works against us! When a battery charges from the grid, it stores that energy in its battery cells, but the conversion from electrical potential in the wires to chemical potential causes a loss of energy. For example, when you charge your phone or other battery-powered electronic device at night, the energy that it draws from the wall is greater than the amount of energy it stores in the battery. Likewise, when you use the device, it needs to discharge more energy from the battery than it takes to power the device. The "loss" of energy can be attributed to the efficiency of the entire system, and energy is usually lost as heat in the components as it travels through them.

Let's define some terms and look at an example wiring diagram of a simple battery.

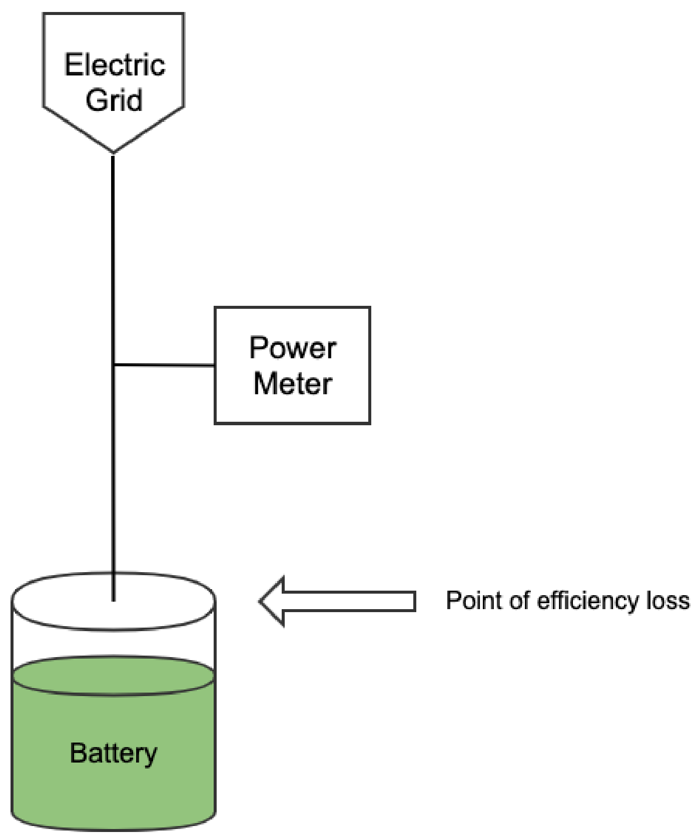
Terms

- **Power (kW):** Electric power is the rate, per unit time, at which electrical energy is transferred by an electric circuit. The SI unit of power is the watt, one joule per second. Electric power is usually produced by electric generators, but can also be supplied by sources such as electric batteries. It is usually supplied to businesses and homes (as domestic mains electricity) by the electric power industry through an electric power grid. In our system, positive values represent discharging power from the battery onto the grid, negative values represent charging the battery from the grid. https://en.wikipedia.org/wiki/Electric_power
- **Energy (kWh):** Electrical energy is usually sold by the kilowatt hour (1 kW·h = 3.6 MJ) which is the product of the power in kilowatts multiplied by running time in hours. Electric utilities measure energy using an electricity meter, which keeps a running total of the electric energy delivered to a customer. https://en.wikipedia.org/wiki/Electrical_energy
- **Capacity (kWh):** Total energy that a battery can hold if fully charged.
- **State-of-charge or SOC (%):** The amount of energy in a battery relative to its total capacity (ex: **80% SOC** means it's storing 80% of the total amount it could store). [*Hint: 80% of the 100 kWh capacity is 80 kWh in this question.]*
- **Efficiency:** Percentage of energy retained during a power exchange in one direction. [*Hint: If you do the math and the numbers don't come out like the filled in table below, you probably forgot to include efficiency in your calculation.]*

In the analogy of "energy" as "water", and a "battery" as a "tank of water", you can think of energy in the battery as the amount of water in a tank. In this analogy, "power" would be the volume of water flowing through a pipe that feeds the tank every given time-step.



[Simple graphical representation of a battery]



[Efficiency loss point compared to where it is measured at the meter]

The following table shows a battery charging and discharging over a two-hour time period (1:00pm to 3:00pm) in half-hour increments. The "power" column is the power read by the "Power Meter" above. The battery's initial SOC is 0.8 (or 80% full):

Time (pm)	Power (KW)	Energy (kWh)	SOC
1:00	0	0	0.80
1:30	50	27.7	0.52
2:00	-25	-11.3	0.63
2:30	-25	-11.3	0.75

Task

Write a simple program in whichever language you prefer (Javascript, Python, LISP, Ada, etc.) to read in the following CSV file, calculate the SOC at each time step, and prints it back out to the terminal/console:

```
Time,Power
1:00,0
1:30,50
2:00,-25
2:30,-25
```

Assume the following:

- Time is in hours (30-min time steps) [Hint: energy is in hours, too!]
- Total Battery Capacity is 100 kWh
- Starting SOC at time 1:00 is 0.80
- Efficiency of moving power in OR out of the battery is 90% (so a 10% loss in each direction)
- Power is read at the meter [*Hint: on charging, efficiency loss happens after the power is read. On discharging, it's the opposite*].
- Power is the average power over that time step.
- You may hard-code the starting SOC, the battery capacity, and efficiency in code (like as global constants).
- You'll need to convert the SOC to the equivalent kWh and back again during this exercise.
- *Hint: It can be easier to write out all of the equations on some paper/whiteboard and try to use time and power to solve for the energy column. Then take the energy lost or gained to solve for the SOC column until you're able to get the same values. This will set you up nicely to convert it into code.*

Questions

1. Start with a single function with the signature: `calculateSoc(hour: float, power: int, currentSoc: float) -> float`
    a. Hour: the length of time, in hours, of the current time step
    b. Power: the rate of energy, in kW, coming into or flowing out of the battery (+ for discharging, - for charging) during this time step.
    c. Beginning SOC: a number between 0 and 1 that represents the SOC of the battery at the beginning of this time step.
    d. Final SOC: the return value of this function, a float between 0 and 1, should be the final SOC at the end of the time step.
2. Expand your program by reading in these values from the above CSV file (called "power_over_time.csv"), feeding in each row into your `calculateSOC()` function, and printing the result to the terminal/console in CSV format:

```
Time,Power,SOC
1:00,0,0.8
1:30,50,0.52
2:00,-25,0.63
2:30,-25,0.75
```

NOTE: You can use an existing CSV library for parsing this, we don't expect you to write your own.

Expected Result

Please return to us the completed single-file code that will read in the CSV from a file named “power_over_time.csv” and prints out the SOC values from the above table. Add any comments and structure that you would normally add to a production code-base. Let us know if you have any questions or get stuck somewhere, we’d be happy to help. Thank you for taking time to do this challenge question for us!