Finance

1a A 500-word explanation of Bitcoin stock-to-flow model and an argument for why it is a bad model

The stock-to-flow model is a bitcoin model popularized by a certain Dutch institutional investor who operates under the Twitter account "PlanB," and has been widely praised as the leading valuation model for bitcoin proponents. Stock-to-Flow model has achieved viral popularity and inspired rags-to-riches dreams for those gambling it all on the future of bitcoin.

PlanB's paper "Modeling Bitcoin Value with Scarcity" states that certain precious metals have maintained a monetary role throughout history because of their unforgeable costliness and low rate of supply. For example, diamond is valuable both because new supply (mined diamond) is insignificant to the current supply and because it is impossible to replicate the vast stores of gold around the globe. PlanB then argues this same logic applies to bitcoin, which becomes more valuable as new supply is reduced every four years, ultimately ending up in a supply of 21 million bitcoin.

Low rate of supply, which PlanB defines as "scarcity," can be quantified using a metric called Stock-to-Flow, which is the ratio between current supply and new supply.

This premise is then translated into the hypothesis, "...that scarcity, as measured by Stock-to-Flow, directly drives value." PlanB then plots bitcoin's Stock-to-Flow against USD market capitalization as well as two arbitrarily chosen Stock-to-Flow data points for gold and silver.

PlanB then runs a linear regression using the natural logarithm of bitcoin's Stock-to-Flow metric as the independent variable and the USD market capitalization as the dependent variable. The paper ends with the conclusion that there is a statistically significant relationship between USD market capitalization and SF values, as evidenced by the linear regression resulting in an R2 (a statistical measure of how close the data fits to a regression line) of ~0.95. The two randomly chosen data points for gold and silver are in line with bitcoin's trajectory and presented as further evidence of the hypothesis.
PlanB suggests that investors can forecast the future USD market cap of bitcoin using the above formula.

Many arguments abound for why this popular model is a bad model. Some of which include;

1. From a theoretical point of view, the model is based on the rather strong assertion that the USD market capitalization of a monetary good (e.g. gold and silver) is derived directly from their rate of new supply. No evidence or research is provided to support this idea, other than the singular data points selected to chart gold and silver's market capitalization against bitcoin's trajectory.

2. The stock-to-flow model has been a novel way of looking at bitcoin's early, meteoric years. However, it will soon break because it predicts nonstop doubling year after year. By 2045, the model estimates each Bitcoin will be worth $235,000,000,000 before eventually converging to infinity as bitcoin's flow approaches 0. Using the estimated slope-intercept formula is making the most naive prediction possible, because bitcoin grew by X in the past, does not necessarily imply it will grow by X in the future. One should remember that past results are not representative of future returns.

1b Calculating the Black-Scholes call price of Yara Inc

(Po)Current price = $40
(X)Strike price =$45
(kRF)Risk free rate = 3%(0.03)
(σ) Volatility = 40%(0.4)
(t)Time to Maturity = 4months(0.33)

d1 = ln(Po/X) +(r +σ^2/2)(t)/σsqrt(t)

= ln(40/45) + (0.03 + 0.4^2/2)0.33/0.4sqrt(0.33)
=-0.08148/0.22978
d1= -0.35

d2= d1 - σsqrt(t)
= -0.35 - 0.4(0.574456)
= -0.35 - 0.22978
= -0.57978
d2= -0.58

Using Standard Normal Distribution Table
N(d1) =0.36
N(d2) =0.28

Black-Scholes call price = SN(d1) - Ke^−rtN(d2)
= 40(0.36) - 45e(−.03(0.3))(0.28)
=40(0.36) - 44.55(0.28)

Black-Scholes call price = $2.02

Computer Science

2a Why it is a bad idea to use recursion method to find the fibonacci of a number

Using recursion method to find the fibonacci of a number is a bad idea because as the nth term parameter(e.g. 67, 100, etc) increases it takes a longer time to generate our fibonacci number. The reason for the poor performance is due to the heavy push-pop of the stack memory in each recursive call.The time complexity: O(n) = O(n-1) + O(n-2) is exponetial.
It is faster and better to use the iterative method instead
Out of curiosity I wrote my own tests using jest and javascript to further research personally why Recursive method is a bad idea for finding the fibonacci of a number. Below are my results which further buttress my answer

```
function fib(n) {
  if (n < 2){
    return n
  }
  return fib(n - 1) + fib (n - 2)
}
```

Recusion Method

Recursive solution:

```
✓ calculates correct fib value for 1 (2ms)
✓ calculates correct fib value for 2 (1ms)
✓ calculates correct fib value for 3 (1ms)
✓ calculates correct fib value for 4
✓ calculates correct fib value for 15 (1328ms)
```

```
function fib(n){
  let arr = [0, 1];
  for (let i = 2; i < n + 1; i++){
    arr.push(arr[i - 2] + arr[i -1])
  }
 return arr[n]
}
```

Iterative Method

Iterative solution:

```
✓ calculates correct fib value for 1 (3ms)
✓ calculates correct fib value for 2 (1ms)
✓ calculates correct fib value for 3 (3ms)
✓ calculates correct fib value for 4 (4ms)
✓ calculates correct fib value for 15 (4ms)
```

2b A function that takes in a Proth Number and uses Proth's theorem to determine if said number is prime

Kindly either click the link below or copy the address to view the live-demo of this answer

https://github.com/AqueousSolution/reasearch_assistant_assessment/blob/master/proth.js

I used javaScript to solve this problem as I am very good with the language and we were also given the freedom to use our preferred programming language.
Although there was a preference for Go and C++, I am not currently well versed in those languages but I have began researching into them to familiarise myself with Buycoin's preferred tools of the trade

In mathematics, a Proth number is a positive integer of the form
$n = k * 2^n + 1$
where k is an odd positive integer and n is a positive integer such that 2n > k

My Approach

1. Deduct 1 from the number. This would give a number in the form k*2n, if the given number is a proth number.
2. Now, loop through all odd number starting form k=1 to n/k and check if k can divide n in such a way that (n/k) is a power of 2 or not.

3a If found, that means our given number is a proth number, so we proceed, in checking if it's also a prime number

3b If no such value of k is found then we terminate the function and Print '**This is not a proth number,please enter a proth number**'

4. We then test the given proth number by dividing it with some given values and check if it those set of values can perfectly divide the given proth number.
5. If the proth number can be further divided by other values asides itself, then we Print 'This number is not a prime number'
6. If the proth number cannot be further divided by other values asides itself, then we Print '**This number is a prime**'

Math

3 Finding the minimum value of a positive real number, y

$y = \sqrt{(x+6)^2 + 25} + \sqrt{(x-6)^2 + 121}$

can also be written as

$y = ((x+6)^2 + 25)^{1/2} + ((x-6)^2 + 121)^{1/2}$

dy/dx = 0 at stationary points therefore differentiate with respect to x

$dy/dx = (x+6)((x+6)^2 + 25)^{-1/2} + (x-6)((x-6)^2 + 121)^{-1/2} = 0$

$$dy/dx = \frac{(x+6)}{((x+6)^2 + 25)^{1/2}} + \frac{(x-6)}{((x-6)^2 + 121)^{1/2}} = 0$$

$$\frac{(x+6)}{((x+6)^2 + 25)^{1/2}} + \frac{(x-6)}{((x-6)^2 + 121)^{1/2}} = 0$$

$$\frac{(x+6)}{((x+6)^2 + 25)^{1/2}} = -\frac{(x-6)}{((x-6)^2 + 121)^{1/2}}$$

get rid of square roots by squaring both sides

$$\frac{(x+6)^2}{((x+6)^2 + 25)} = \frac{(x-6)^2}{((x-6)^2 + 121)}$$

cross multiply both sides

$(x+6)^2(x-6)^2 + 121(x+6)^2 = (x-6)^2(x+6)^2 + 25(x-6)^2$

deduct $(x+6)^2(x-6)^2$ from both sides

$121(x+6)^2 = 25(x-6)^2$

$121x^2 + 1452x + 4536 = 25x^2 - 300x + 900$

collect like terms together

$96x^2 + 1752x + 3456 = 0$

this can be further simplified by dividing through by 24

$4x^2 + 73x + 144 = 0$

using the almighty formula, solve for x

$$\frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x = \frac{-73 \pm \sqrt{73^2 - 4(4)(144)}}{2(4)}$$

$$x = \frac{-73 \pm 55}{8}$$

$$x = -16 \text{ or } -2.25$$

when $x = -16$, $y = \sqrt{(-16+6)^2 + 25} + \sqrt{(-16-6)^2 + 121}$

$$y = 35.77$$

when $x = -2.25$, $y = \sqrt{(-2.25+6)^2 + 25} + \sqrt{(-2.25-6)^2 + 121}$

$$y = 20$$

This implies that the minimum value of y is 20