1️⃣

Math 449: Assignment 1

Question 1

Consider the number $\phi = \frac{\sqrt{5}-1}{2}$

Part A)

Show that for all integers $n \geq 0$ we have $\phi^{n+2} = \phi^n - \phi^{n-1}$. Deduce that starting with $\phi^0 = 1, \phi^1 = \phi$, all powers ϕ^n can be computed using the recurrence relation $x_{n+2} = x_n - x_{n+1}$, without invoking the multiplication operation.

Base case:

$$\phi^2 = (\frac{\sqrt{5}-1}{2})^2 = \frac{6-2\sqrt{5}}{4} = \frac{3-\sqrt{5}}{2} = 1 + \frac{1-\sqrt{5}}{2} = \phi^0 - \phi$$

Let $k \in \mathbb{N}, k \geq 2$, and we want to prove that $\phi^{k+2} = \phi^k - \phi^{k+1}$ being true for k implies that it also holds for $k+1$.

$$\phi^{k+3} = \phi^{k+2}\phi$$

We have from our induction hypothesis that $\phi^{k+2} = \phi^k - \phi^{k+1}$, so we also have that

$$\phi^{k+3} = \phi\phi^{k+2} = \phi(\phi^k - \phi^{k+1}) = \phi^{k+1} - \phi^{k+2}$$

Which shows that if $\phi^{k+2} = \phi^k - \phi^{k+1}$ is true for some k, it is also true for $k+1$, and therefore via induction it is true for all $k \in \mathbb{N}, k \geq 2$.

Part B)

Write a small computer code (using double precision floating-point arithmetic, e.g. in Matlab or python) that computes the powers of φ based on the established recurrence relation. Use your code to compute all powers of φ up to n = 100.

Code for computing powers of ϕ (full notebook attached with assignment PDF):

```
results = []
phi = np.float64((np.sqrt(5) - 1) / 2)
results.append(phi**0)
results.append(phi**1)
```

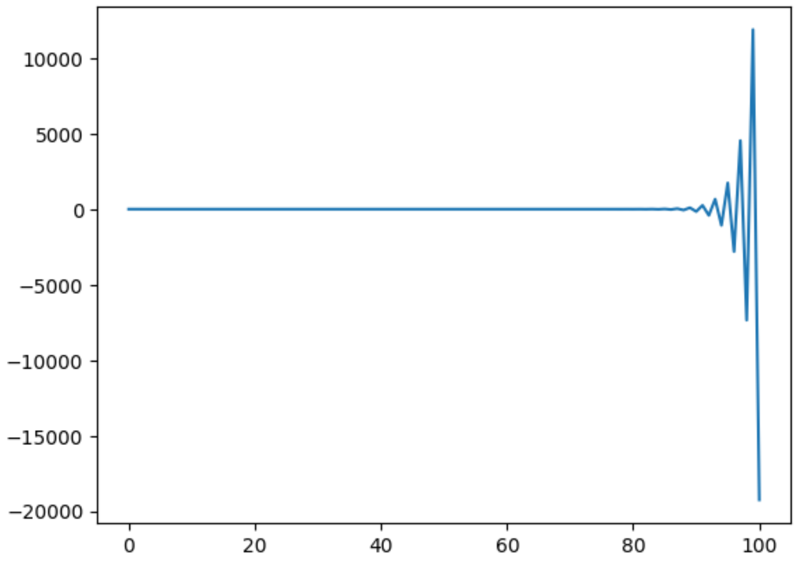
```
for i in range(2, 101):
    new_phi = results[-2] - results[-1] # phi**(i+2) = phi**(i) - phi**(i+1)
    results.append(new_phi)

print(results)
```

Display or plot the resulting sequence and discuss what you observe. In particular, plot the natural log of the absolute value of the computed sequence. Are
the results meaningful to you? Explain.

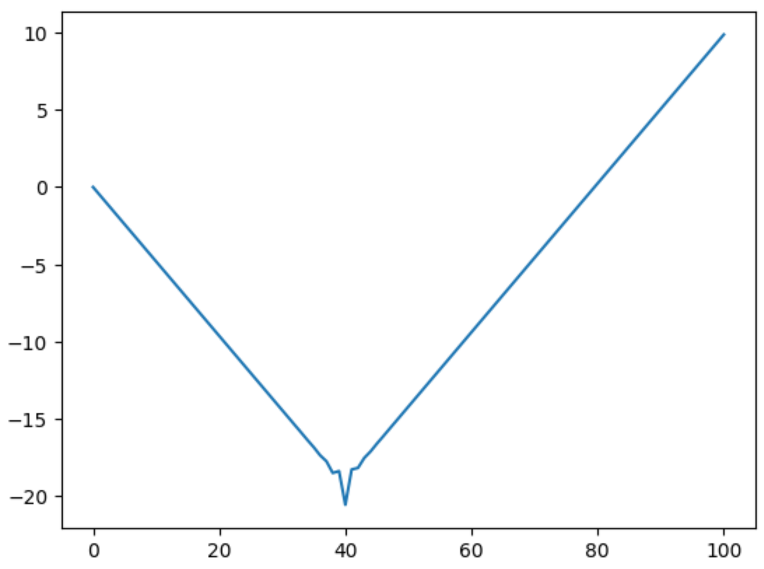
Raw values:

```
plt.plot(results)
plt.show()
```



Log of absolute values:

```
log_results = [np.log(np.abs(x)) for x in results]
plt.plot(log_results)
plt.show()
```



Here we can see that everything is working as intended for the first ~40ish powers (the value exponentially approaches 0). Then we transition into the subnormal numbers briefly, and then we have an overflow and the value begins to grow exponentially.

Part C)

Note that ϕ is a root of the quadratic equation $x^2+x-1=0$. Find the other root and denote the two roots as ϕ_+ and ϕ_-.

We have the following via the quadratic formula:

$$x^2+x-1=0 \to x = \frac{-b\pm\sqrt{b^2-4ac}}{2a} = \frac{-1\pm\sqrt{1+4}}{2} = \frac{-1\pm\sqrt{5}}{2}$$

So our two roots are $\phi_+ = \frac{-1+\sqrt{5}}{2}, \phi_- = \frac{-1-\sqrt{5}}{2}$.

Part D)

Consider an arbitrary sequence with arbitrary given initial values x_0, x_1. Assume that $x_0 = c_1 + c_2, x_1 = c_1\phi_+ + c_2\phi_-$, show that $x_n = c_1\phi_+^n + c_2\phi_-^n$.

Our facts to begin with are:

$$x_n = x_{n-2} - x_{n-1}, x_0 = c_1 + c_2, x_1 = c_1\phi_+ + c_2\phi_-$$

Therefore, using the fact that $\phi_+^{n+2} = \phi_+^n - \phi_+^{n-1}, \phi_-^{n+2} = \phi_-^n - \phi_-^{n-1}$, we have:

$$x_2 = (c_1+c_2) - (c_1\phi_+ + c_2\phi_-) = (c_1 - c_1\phi_+) + (c_2 - c_2\phi_-) = c_1(1-\phi_+) + c_2(1-\phi_-) = c_1\phi_+^2 + c_2\phi_-^2$$

And similarly to above, we can prove that this will apply to all $n \geq 2, n \in \mathbb{N}$, using the above as our base case.

$$x_{n+1} = x_{n-1} - x_n = (c_1\phi_+^{n-1} + c_2\phi_-^{n-1}) - (c_1\phi_+^n + c_2\phi_-^n) = c_1(\phi_+^{n-1} - \phi_+^n) + c_2(\phi_-^{n-1} - \phi_-^n) = c_1\phi_+^{n+1} + c_2\phi_-^{n+1}$$

Therefore we have via induction that:

$$x_n = c_1\phi_+^n + c_2\phi_-^n$$

For all $n \geq 2, n \in \mathbb{N}$.

Solve for c_1, c_2 given x_0, x_1.

Given that:

$$x_0 = c_1 + c_2, x_1 = c_1\phi_+ + c_2\phi_-$$

We have that:

$$c_1 = c_2 - x_0$$

Therefore:

$$x_1 = (c_2 - x_0)\phi_+ + c_2\phi_- = c_2\phi_+ - x_0\phi_+ + c_2\phi_- \to c_2(\phi_+ + \phi_-) = x_1 - x_0\phi_+ \to c_2 = \frac{x_1 - x_0\phi_+}{\phi_+ + \phi_-}$$

And now we can see that

$$c_1 = c_2 - x_0 \to c_1 = \frac{x_1 - x_0\phi_+}{\phi_+ + \phi_-} - x_0$$

Part E)

Consider $x_0 = 1, x_1 = \phi_+ + \epsilon, \epsilon = 10^{-16}$. Find c_1, c_2 such that $x_n = c_1\phi_+^n + c_2\phi_-^n$.

Using what we've derived above, we have:

$$c_1 = \frac{(\phi_+ + \epsilon) - \phi_+}{\phi_+ + \phi_-} - 1 = \frac{\epsilon}{\phi_+ + \phi_-} - 1 = \frac{10^{-16}}{-1} - 1$$

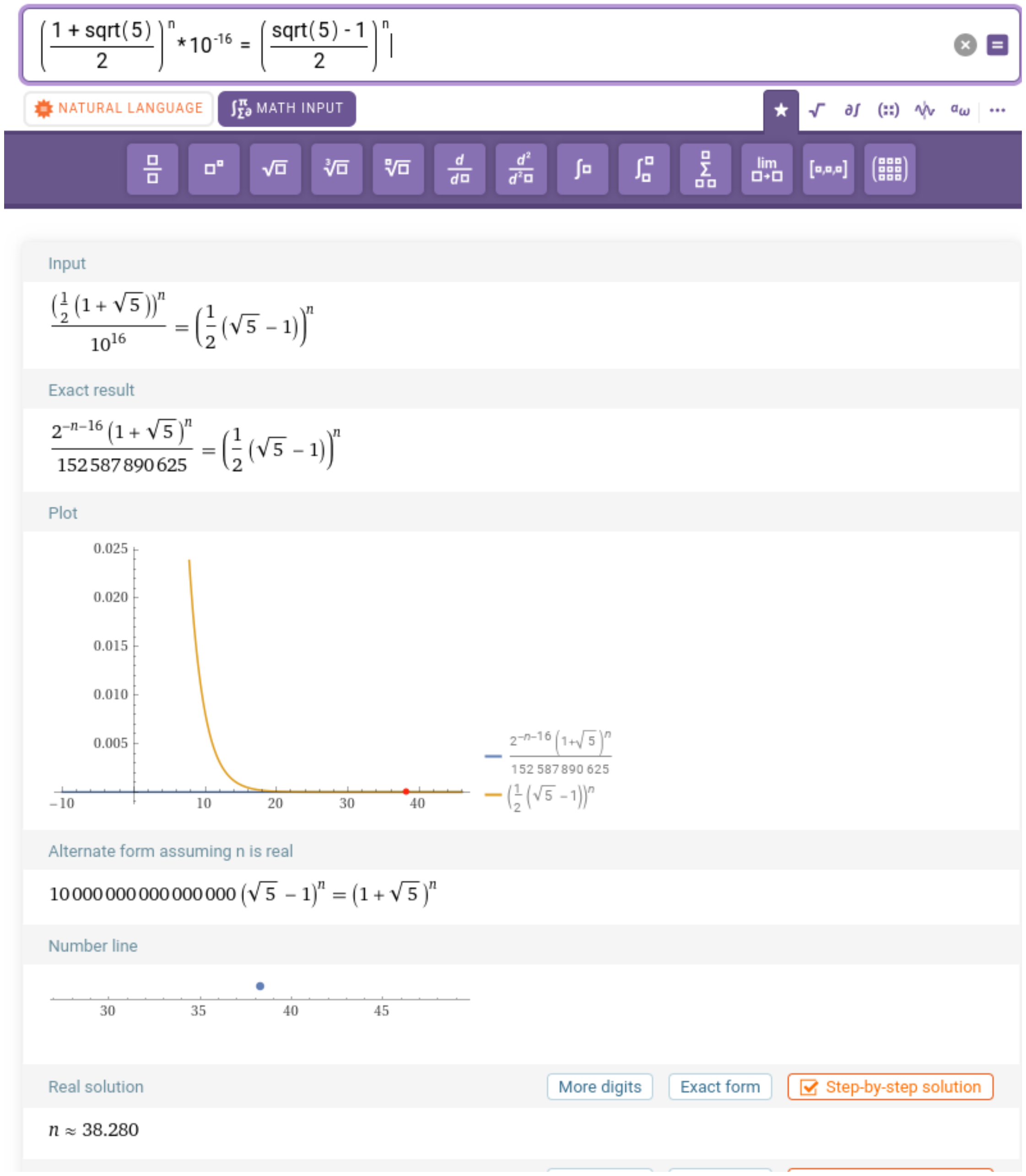
$$c_2 = \frac{(\phi_+ + \epsilon) - \phi_+}{\phi_+ + \phi_-} = \frac{\epsilon}{\phi_+ + \phi_-} = -10^{16}$$

Then justify the results of the numerical simulation from earlier.

Our c_2, instead of being 0, is a number very close to 0. That means that when we calculate $x_n = c_1\phi_+^n + c_2\phi_-^n$ the ϕ_-^n can eventually get large enough to "dominate" the other terms and x_n grows exponentially. This also explains why we see the transition at $n = 40$, as that is when $\phi_-^n(\frac{\epsilon}{\phi_+ + \phi_-})$ gets close to ϕ_+^n. In fact, using WolframAlpha, if we input the equality (which is not quite the same, but behaves very similarly and allows us to solve for a real n instead of a complex n):

$$(\frac{1+\sqrt{5}}{2})^n 10^{-16} = (\frac{\sqrt{5}-1}{2})^n$$

We get $n \approx 38$.



$$\left(\frac{1+\text{sqrt}(5)}{2}\right)^n * 10^{-16} = \left(\frac{\text{sqrt}(5)-1}{2}\right)^n$$

Input

$$\frac{\left(\frac{1}{2}\left(1+\sqrt{5}\right)\right)^n}{10^{16}} = \left(\frac{1}{2}\left(\sqrt{5}-1\right)\right)^n$$

Exact result

$$\frac{2^{-n-16}\left(1+\sqrt{5}\right)^n}{152\,587\,890\,625} = \left(\frac{1}{2}\left(\sqrt{5}-1\right)\right)^n$$

Alternate form assuming n is real

$$10\,000\,000\,000\,000\,000\left(\sqrt{5}-1\right)^n = \left(1+\sqrt{5}\right)^n$$

Real solution

$n \approx 38.280$

Question 2

Consider the following:

```
x = 0.5
while 1+x > 1:
    x /= 2
print(x)
```

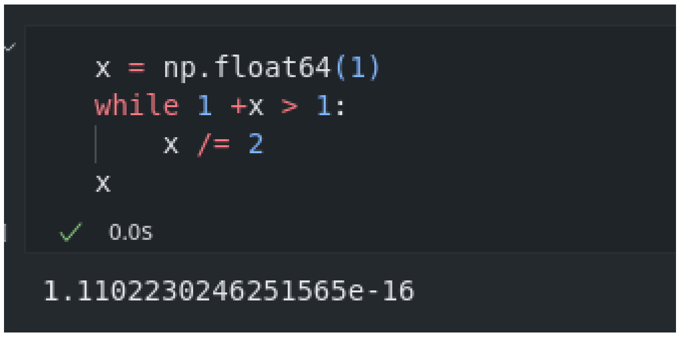
Part A)

Why does this code give us an estimate of the machine epsilon?

The machine epsilon is defined as the smallest number ϵ such that $1 + \epsilon > 1$. In our code the while condition is this condition, so when the while condition becomes false x will be half the machine epsilon

Part B)

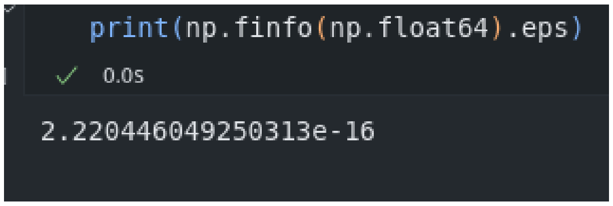
Implement the above pseudo-code to get an estimate of your machine's epsilon.



```
x = np.float64(1)
while 1 +x > 1:
    x /= 2
x
```

```
0.0s
1.1102230246251565e-16
```

And here we can see that this is about half of the machine epsilon we get from directly querying NumPy:



```
print(np.finfo(np.float64).eps)
```

```
0.0s
2.220446049250313e-16
```

Question 3)

Let $||\cdot||_v$ be a vector norm of $\mathbb{R}^n$ and for any $n \times n$ matrix A let

$$||A||_M = max_{X!=0} \frac{||AX||_v}{||X||_v}$$

Prove that $||\cdot||_M$ is a matrix norm.

To prove that this is a matrix norm, we can simply prove that it satisfies all the qualities that matrix norms are defined by:

1. $||A||_M \geq 0$
2. $||A||_M = 0 \leftrightarrow A = 0$
3. $||cA||_M = |c|||A||_M$

4. $||A+B||_M \leq ||A||_M + ||B||_M$
5. $||AB||_M \leq ||A||_M ||B||_M$

i) Showing that $||\cdot||_M$ is non-negative.

Because $||\cdot||_v$ is a vector norm, it is always non-negative. Therefore $||AX||_v$ is non-negative.

ii) Showing that $||\cdot||_M$ is definite.

First we can observe that if $A=0$, then $AX=0$ for all X. Therefore we have that $A=0 \to ||A||_M = 0$.

Then in the other direction we can see if $||A||_M = 0$ then that means that $||AX||_v = 0$ for all X, which means that A must be the matrix with all 0 entries. Therefore we have that $||A||_M = 0 \to A = 0$, and from here we can combine both of these into

$$A = 0 \leftrightarrow ||A||_M = 0$$

iii) Showing preservation of scalar multiplication.

$$||cA||_M = max_{||X||_v != 0} \frac{||cAX||_v}{||X||_v} = |c|(max_{||X||_v != 0} \frac{||AX||_v}{||X||_v}) = |c|||A||_M$$

iv) Triangle inequality

$$||A+B||_M = max_{||X||_v != 0} \frac{||(A+B)X||_v}{||X||_v} = max_{||X||_v != 0} \frac{||AX+BX||_v}{||X||_v} \leq max_{||X||_v != 0} \frac{||AX|| + ||BX||_v}{||X||_v}$$

We have the last inequality here because $||\cdot||_v$ is a vector norm and itself must satisfy the triangle inequality.

v) Sub-multiplicative Property

$$||AB||_M = max_{||X||_v != 0} \frac{||ABX||_v}{||X_v||} = max_{||BX||_v \neq 0} \frac{||ABX||_v}{||X||_v} =$$

$$max_{||BX||_v \neq 0} \frac{||ABX||_v}{||BX||_v} \frac{||BX||_v}{||X||_v} \leq (max_{||Y||_v \neq 0} \frac{||AY||_v}{||Y||_v})(max_{||X||_v \neq 0} \frac{||BX||_v}{||X||_v}) = ||A||_M ||B||_M$$

Question 4)

Prove:

$$||A||_\infty = max_{1 \leq j \leq n} \Sigma_{i=1}^n |a_{ij}|$$

First lets write down the definition of $||A||_\infty$ matrix norm:

$$||A||_\infty = max_{||X||_\infty=1}||AX||_\infty$$

And the definition of the $||X||_\infty$ vector norm:

$$||X||_\infty = max_{1\leq i\leq n}|x_i|$$

And matrix-vector multiplication:

$$(AX)_i = \Sigma_{j=1}^n a_{ij}x_i$$

From these definitions we can show what we want.

$$||A||_\infty = max_{||X||_\infty=1}||AX||_\infty = max_{||X||_\infty=1}(max_{1\leq i\leq n}|(AX)_i|) = max_{||X||_\infty=1}(max_{i\leq 1\leq n}|\Sigma_{j=1}^n a_{ij}x_i|)$$

And because we have that $||X||_\infty = 1$ (that is, that the largest x_i is 1, we have:

$$||A||_\infty = max_{||X||_\infty=1}(max_{i\leq 1\leq n}|\Sigma_{j=1}^n a_{ij}x_i|) = max_{i\leq 1\leq n}|\Sigma_{j=1}^n a_{ij}|$$