Question 1

Inputs:

A, B, S: The coordinates of points A, B, and S in a 3D space

v: Speed of car

V: Speed of bullet

Output:

- Coordinates of point C
- "NO" if no point exists

Key Points

- Car moves in a straight line from point A to B at a constant speed v

- The car's position at any time t can be expressed as point C

- For the car & bullet to meet at point C, the time it takes for the car to reach C must be equal to the time for the bullet to reach C

Question 2

To compute cos(x) exactly using the basic calculator operations and following numbers, it can be done as shown below:

1) $A(n) = \cos(Nx)$

The calculation of $\cos(Nx)$ for any integer N can be efficiently done using the Chebyshev polynomials. The formula below can be used recursively:

$$\cos((k+1)x) = 2\cos(x)\cos(kx) - \cos((k+1)x)$$

This allows us to compute $\cos(Nx)$ for any N. The # of operations scales linearly with N since the program would recursively go from $\cos(2x), \cos(3x), \cos(4x), \ldots, \cos(Nx)$

Therefore, the time complexity is $O(N)$

2) $B(N) = \cos\left(\frac{\pi}{2N}\right)$

This can be approached by using the half angle formula:

$$\cos\left(\frac{x}{2}\right) = \sqrt{\frac{1+\cos(x)}{2}}$$

This formula can also be applied recursively:

$$\cos\left(\frac{x}{2^2}\right) = \cos\left(\frac{x}{4}\right) = \sqrt{\frac{1+\cos\left(\frac{x}{2}\right)}{2}}$$

This can be repeated until $\cos\left(\frac{x}{2^N}\right)$ which has a time complexity of $O(N)$

3) $\cos\left(\frac{x}{3}\right)$

This can be computed using the triple angle formula:

$$\cos(3\theta) = 4\cos^3\theta - 3\cos(\theta)$$

For $\theta = \frac{x}{3}$, it can be written as:

$$\cos(x) = \cos\left(3\cdot\frac{x}{3}\right) = 4\cos^3\left(\frac{x}{3}\right) - 3\cos\left(\frac{x}{3}\right)$$

Let $y = \cos\left(\frac{x}{3}\right)$. The equation becomes:

$$\cos(x) = 4y^3 - 3y$$

It can then be written as a cubic equation

$$4y^3 - 3y - \cos(x) = 0$$

Question 3

The recursion relation involves one division operation for each step from $x(i)$ to $x(i+1)$. Therefore, to compute $x(n)$, you will need to perform exactly n division operations

The time complexity would be $O(n)$. Computing $x(2^{1000})$ is possible but in practice, it would take a long time

Question 4

To take the determinant of a nxn matrix, you can use expansion by minors (Laplace expansion

This method requires expanding the determinant along a row or column. For an $n \times n$ matrix $A = (a_{ij})$, the determinant is calculated as shown below:

$$\det(A) = \sum_{j=1}^{n} (-1)^{i+j} a_{ij} \det(M_{ij}$$

Where M_{ij} is the $(n-1) \times (n-1)$ submatrix obtained by removing the i^{th} row and j^{th} from A