Questions:

1. Write a program in your favourite procedural language that does linear search in an array

2. The logical formula \( p \rightarrow \neg q \) is equivalent to which of the following:

   - [A] \( \neg p \rightarrow q \)
   - [B] \( \neg (p \rightarrow q) \)
   - [C] \( q \rightarrow \neg p \)
   - [D] \( \neg q \lor \neg p \)

3. Prove by induction that \( n! \geq 2^{n-1} \) for \( n \geq 1 \)

Answers:

1. Below is a linear search for a key in an array, written in the C language.

```c
/*
   Does array 'a' of length 'len' contain the key 'val'? If true, return the index of the key. If false, return -1
*/
int LinearSearch(int a[], int len, int val)
{
    int found = -1; // assume not found
    for (int i=0; i<len && found==-1; i++)
    {
        if (a[i] == val) found = i; // found the key
    }
    return found;
}
```

2. The logical formula \( p \rightarrow \neg q \) is equivalent to [C] \( q \rightarrow \neg p \), and to [D] \( \neg q \lor \neg p \)

3. Proof by induction that \( n! \geq 2^{n-1} \) for \( n \geq 1 \).

   **Base case:** \( n = 1 : 1! \geq 2^0 \) is true

   **Inductive hypothesis:** assume true for \( n = k : k! \geq 2^{k-1} \)

   **Inductive step:** prove true for \( n = k + 1 \), that is, prove \( (k+1)! \geq 2^k \)

   \[
   (k+1)! = (k+1) \times k! \geq (k+1) \times 2^{k-1} \geq 2 \times 2^{k-1} = 2^k \quad \text{property of factorials}
   \]

   substitute IH into previous line

   \[
   k + 1 \geq 2 \text{ for } k \geq 1, \text{ property of exponents}
   \]

   QED