1. What data type in the C programming language allows for the largest values of factorial to be computed?
The largest data type is “long long” data type which guarantees at least 64 bits. Compared to int type which allows 32 bits. The range of value allows for long long data type if the value is unsigned is from 0 ~ 18446744073709551615(2^64 – 1). The range of value allows for int data type if the value if unsigned is from 0 ~ 4294967295(2^32 -1). 2^32 -1 is approximately between 12 factorial and 13 factorial.

2. At what input value for the recursive factorial function does your computer start to 'crash' or really slow down when you try to compute a factorial? Is it the same value as the iterative function? Experiment and report your result.
2^32 -1 is approximately between 12 factorial and 13 factorial. When the input is beyond 12, the int type can’t cover the value. 2 ^ 64 -1 is approximately between 20 factorial and 21 factorial. When the input is beyond 20 the long long type can’t cover the result value.
In term of time, program does not slow down. When the input is above 60, the result will be 0.

3. In 2-3 sentences, describe why you believe you saw or didn't see differences between the iterative and recursive versions of factorial.
When the input is below 20, the result is correct. When it goes above 21, it will produce incorrect answer. When the input is above 70, the result will be 0. When the input is above 1000, the recursive way will crash due to stack over flow.