Problem 1

(a)

Yes. Because this programs continues when a != 'y' && a != 'Y', so when a == 'y' or a =='Y', this loop will be terminated. The logic operator should be &&, because if it's ||, this program continues when the user enters a number that's not 'y' or not 'Y', basically all characters satisfies this condition, even when a == 'y', it satisfies the statement a != 'Y', and it satisfies the statement a != 'y' when a == 'Y',so this program won't terminates with any input.

(b)

The programs still takes every characters individually, if there's a 'y' or 'Y' in one line of input, the loop will still be terminated.

Because when the user inputs a serial of numbers in the same time with a keyboard, they're first stored into the keyboard buffer, then the object cin will repeatedly read the first character in the buffer if there's anything in it, so entering a serial of characters at the same time is the same with entering them individually, the only thing that differs would be if there's a 'y' or 'Y' in the input the loop will be terminated immediately, all characters behind the first 'y' or 'Y' will be ignored and left in the buffer.

Problem 2

By the standard definition in C++, i = i+1; is equivalent to ++i; because they both represents the following steps:

1. Take out the value from the address with the identifier i in the memory.
2. Store the value to the register.
3. Add 1 to the value in the register.
4. Store the value in the register back to the original address.

But because the return values of the two statements ++i; and i++; are both ignored, so in most modern compilers, they're handled in the same way like ++i; which takes less machine cycles to complete after compiler optimization.