cin >> ch;
}
cout << "The value of times is " << times << "\n";
cout << "Bye\n";
return 0;
}

void n_chars(char c, int n) // displays c n times
{
    while (n-- > 0)         // continue until n reaches 0
        cout << c;
}

Here is a sample run:

Enter a character: W
Enter an integer: 50
WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
Enter another character or press the q-key to quit: a
Enter an integer: 20
aaaaaaaaaaaaaaaaaaaa
Enter another character or press the q-key to quit: q
The value of times is 20.
Bye

Program Notes

The main() function uses a while loop to provide repeated input (and to keep your loop skills fresh). Note that it uses cin >> ch to read a character rather than cin.get(ch) or ch = cin.get(). There's a good reason for doing so. The cin.get() pair of functions, you recall, read all input characters, including spaces and newlines, whereas cin >> skips spaces and newlines. When you respond to the program prompt, you have to press Enter at the end of each line, thus generating a newline character. The cin >> ch approach conveniently skips over these newlines, but the cin.get() siblings read the newline following each number entered as the next character to display. You can program around this nuisance, but it's simpler to use cin as the program does.