Indefinite Loop Categories

Indefinite loops are those that wait until an event occurs at runtime to complete. You might wonder, "what kind of event could that be?". Here are three kinds of indefinite bounds; each uses a different sort of bounds:

Data bound loops are those that keep **reading from input** until there is **no more data** to be read. Data loops are used when processing files or network data. We'll work extensively with data loops when we get to **streams**.

Here is an indefinite data loop that reads all of the words from a file, represented by the input file object named `in`, and prints each one on its own line:

```
string word;
while (in >> word)
{
    cout << word << endl;
}
```

Sentinel bound loops look for the presence of a special value, **contained within its input**, to determine when to quit. If your problem is "read characters until you encounter a period", then the period is the sentinel.

Sentinel loops are often used in searching, but have other uses as well. This sentinel loop finds the position of the first period entered.

```
int position = 0; char c;
cin >> c;
while (c != '.') // Loop sentinel
{
    position++;
    cin >> c;
}
```

Limit bound loops end when another repetition of the loop won't get you any closer to your goal. They are often used in scientific calculations and other numeric algorithms, when stating a precise termination condition is not possible.

Often this involves monitoring the difference between two variables, and stopping the loop when the difference passes a predetermined threshold. Here is a limit-loop example which counts the number of odd digits in an integer `n`:

```
int count = 0;
while (n != 0) // Loop limit
{
    if (n % 10 % 2 == 1) { count++; }
    n = n / 10;
}
```