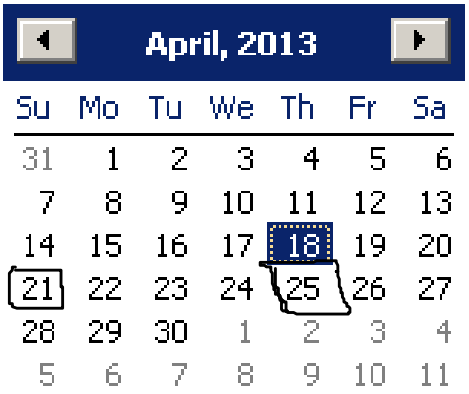
engi231020p2c3

Thursday, April 18, 2013
4:40 PM

S6: 14 abril	• Clases 8 y 9 8. Techniques in the Solution of Problems 9. Sequential Structures
S7: 21 abril	• Clases 10. Decision Structures • **Segunda Prueba Parcial**



La semana del 21 de abril, en particular el jueves 25 de abril daremos un test como el primero (multichoice de 8 o 10 preguntas)

Hasta ahora hemos visto:

- Instrucciones secuenciales.
- Instrucciones condicionales (if else).
- Instrucciones cíclicas (for, while).

5.2 Introduction to Loops: The `while` Loop

CONCEPT: A loop is part of a program that repeats.

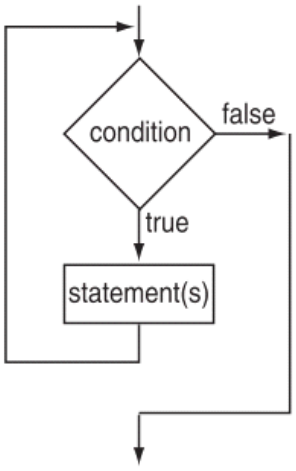
Chapter 4 introduced the concept of control structures, which direct the flow of a program. A *loop* is a control structure that causes a statement or group of statements to repeat. C++ has three looping control structures: the `while` loop, the `do-while` loop, and the `for` loop. The difference between each of these is how they control the repetition.

The `while` Loop

The `while` loop has two important parts: (1) an expression that is tested for a true or false value, and (2) a statement or block that is repeated as long as the expression is true. Figure 5-1 shows the general format of the `while` loop and a flowchart visually depicting how it works.

Figure 5-1

```
while (condition)
{
     statement;
     statement;
     // Place as many statements
     // here as necessary
}
```



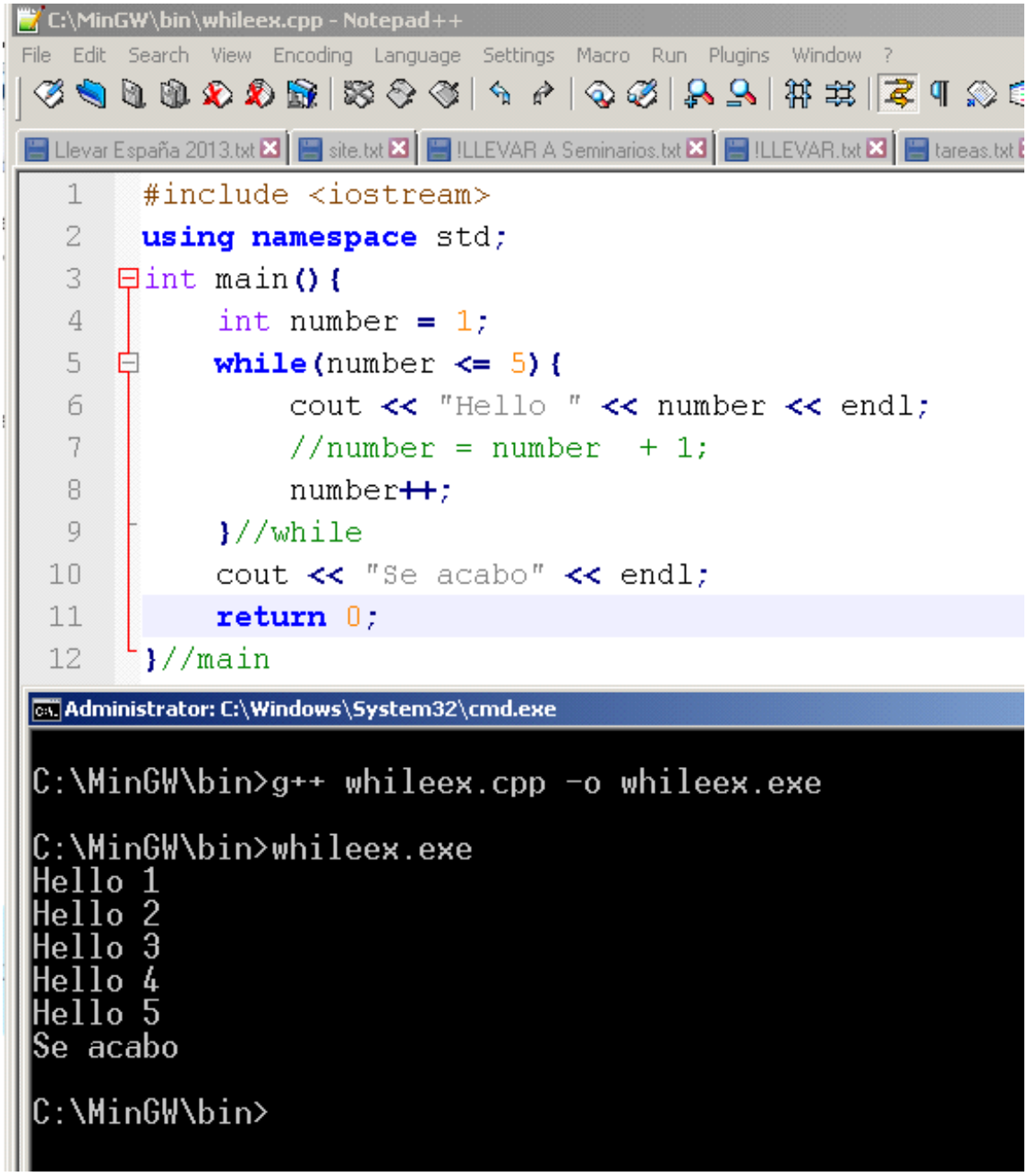
Let's look at each part of the `while` loop. The first line, sometimes called the *loop header*, consists of the key word `while` followed by a *condition* to be tested enclosed in parentheses.

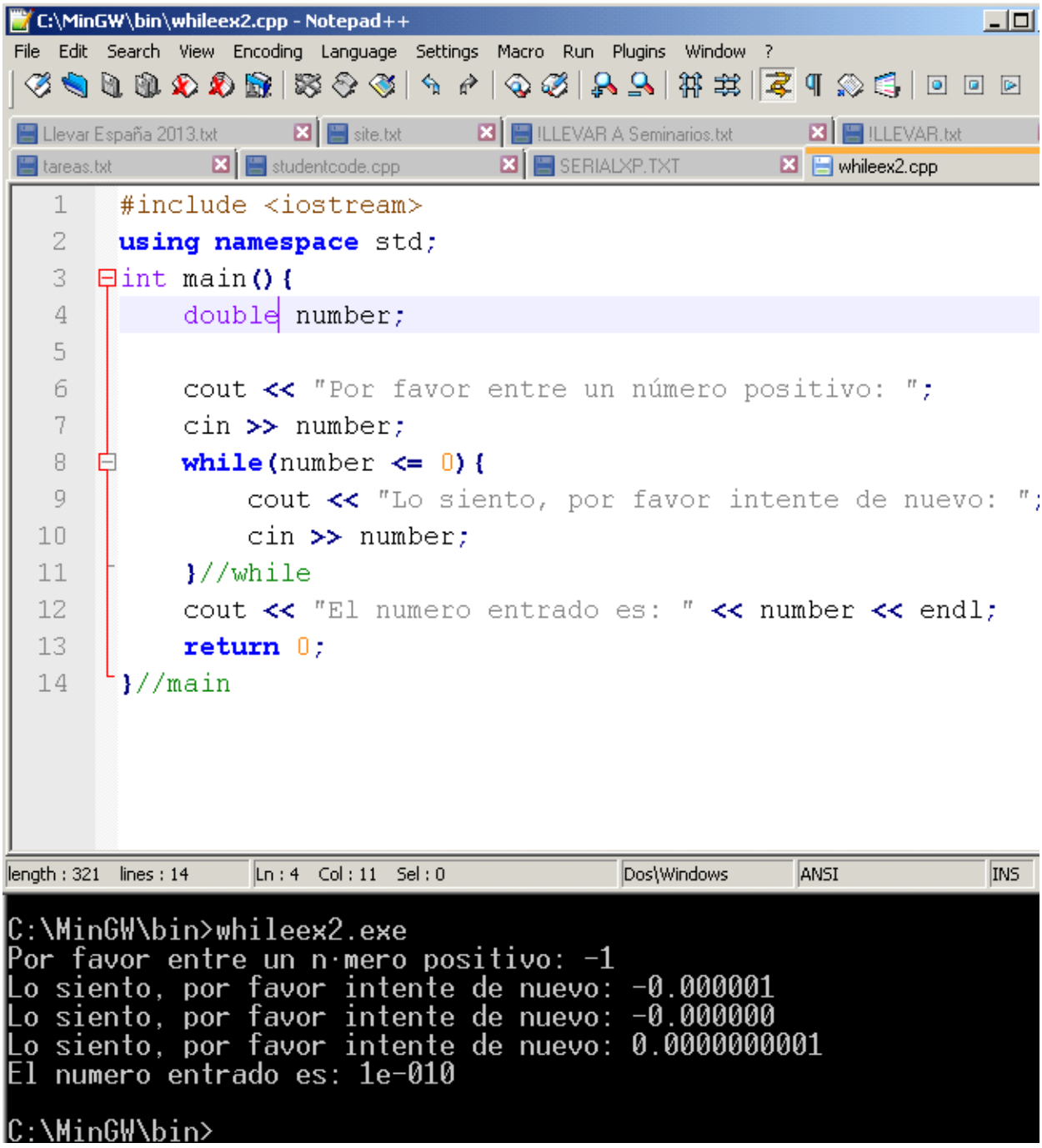
Program 5-3

```
1  // This program demonstrates a simple while loop.
2  #include <iostream>
3  using namespace std;
4
5  int main()
6  {
7     int number = 1;
8
9     while (number <= 5)
10    {
11       cout << "Hello   ";
12       number++;
13    }
14    cout << "\nThat's all!\n";
15    return 0;
16 }
```

Program Output

```
Hello   Hello   Hello   Hello   Hello
That's all!
```


C:\MinGW\bin\whileex.cpp - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
Llevar España 2013.txt
site.txt
!LLEVAR A Seminarios.txt
!LLEVAR.txt
tareas.txt
#include <iostream>
using namespace std;
int main(){
int number = 1;
while(number <= 5){
cout << "Hello " << number << endl;
//number = number + 1;
number++;
}//while
cout << "Se acabo" << endl;
return 0;
}//main
Administrator: C:\Windows\System32\cmd.exe
C:\MinGW\bin>g++ whileex.cpp -o whileex.exe
C:\MinGW\bin>whileex.exe
Hello 1
Hello 2
Hello 3
Hello 4
Hello 5
Se acabo
C:\MinGW\bin>



Validacion de datos por si entra un caracter o numeros decimales:

```cpp
using namespace std;
int main(){
    int number;
    bool fTexto;
    cout << "Por favor entre un numero positivo: ";
    cin >> number;
    fTexto = cin.fail();
    cout << "fTexto: " << fTexto; //Esto es true cuando hay texto!!!
    cout << "number: " << number;
    while(fTexto || number <= 0){
        cout << "number: " << number << endl;
        if(fTexto) cout << "Usted entro un caracter no numerico. " << endl;
        else cout << "Usted entro un numero no positivo. " << endl;
        cout << "Lo siento, por favor intente de nuevo: ";
        cin.clear();
        cin.ignore(80, '\n');
        cin >> number;
        fTexto = cin.fail();
        cout << "fTexto: " << fTexto;
    }//while
    cout << "fTexto: " << fTexto;
    cout << "El numero entrado es: " << number << endl;
    return 0;
}//main
```

Si el usuario entra un caracter (a, A, etc) cin.fail() es true y entra dentro del while y tambien dentro del if y le da mensaje al usuario.

Si el usuario entra un decimal, cin.fail() es false (no nos vale de nada) pero number tiene el valor truncado. Al llegar a cin.clear() y cin.ignore(….); limpia el resto y permite que el usuario vuelva a entrar datos sin volverse loca la compu.

El siguiente código mira primero si no es texto y luego si no es negativo y tiene en cuenta si el usuario mete un numero decimal, lo trunca y limpia el buffer para que la compu no se vuelva loca

```
int main(){
    int number;
    bool fTexto;
    cout << "Por favor entre un numero positivo: ";
    cin >> number;
    fTexto = cin.fail();
    while(fTexto){
        cout << "Usted entro un caracter no numerico. " << endl;
        cout << "por favor intente de nuevo: ";
        cin.clear();
        cin.ignore(80, '\n');
        cin >> number;
        fTexto = cin.fail();
    }//while

    while(number <= 0){
        cout << "Usted entro un numero no positivo. " << endl;
        cout << "por favor intente de nuevo: ";
        cin.clear();
        cin.ignore(80, '\n');
        cin >> number;
    }//while

    cout << "El numero entrado es: " << number << endl;
```