

<b>S4: Lunes 31 marzo</b>	<ul style="list-style-type: none"> <li>● <i>Basic input and output (cin, cout, &gt;&gt;, &lt;&lt;, get, put)</i></li> <li>● <b>Ejemplos usando tipo de datos enteros y flotantes.</b></li> </ul>
<b>S5: Lunes 7 abril</b>	<ul style="list-style-type: none"> <li>● <i>Problem solving methods (algorithms, flowcharts).</i></li> </ul>

**Program 3-23**

```

1 // This program reads a single character into a char variable.
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     char ch;
8
9     cout << "Type a character and press Enter: ";
10    cin >> ch;
11    cout << "You entered " << ch << endl;
12    return 0;
13 }

```

**Program Output with Example Input Shown in Bold**

```

Type a character and press Enter: A[Enter]
You entered A

```

Pero cuando uno entra blancos, el cin >> ch no funciona. En ese caso se usa el cin.get();

**Program 3-24**

```

1 // This program demonstrates three ways
2 // to use cin.get() to pause a program.
3 #include <iostream>
4 using namespace std;
5
6 int main()
7 {
8     char ch;
9
10    cout << "This program has paused. Press Enter to continue.";
11    cin.get(ch);
12    cout << "It has paused a second time. Please press Enter again.";
13    ch = cin.get();
14    cout << "It has paused a third time. Please press Enter again.";
15    cin.get();
16    cout << "Thank you!";
17    return 0;
18 }

```

**Program Output with Example Input Shown in Bold**

```

This program has paused. Press Enter to continue.[Enter]
It has paused a second time. Please press Enter again.[Enter]
It has paused a third time. Please press Enter again.[Enter]
Thank you!

```

## Ejemplo con cin.get

```
3 int main(){
4     char myChar;
5     cout << "Bienvenidos a CECS2200, entre una letra: ";
6     //cin >> myChar;
7     cin.get(myChar);
8     cout << "La letra entrada es <" << myChar << ">" << endl;
9     return 0;
10 }
```

```
C:\MinGW\bin>main2
Bienvenidos a CECS2200, entre una letra: x
La letra entrada es <x>
```

```
C:\MinGW\bin>main2
Bienvenidos a CECS2200, entre una letra:
La letra entrada es < >
```

```
C:\MinGW\bin>main2
Bienvenidos a CECS2200, entre una letra:
La letra entrada es <
>
```

← es enter

← Le di space

← Le di

Uso de put: esta instrucción se usa para escribir caracteres en un fichero de texto. Ejemplo:

```
1 // typewriter
2 #include <iostream> // std::cin, std::cout
3 #include <fstream> // std::ofstream
4
5 int main () {
6     std::ofstream outfile ("test.txt");
7     char ch;
8     std::cout << "Type some text (type a dot to finish):\n";
9     do {
10        ch = std::cin.get();
11        outfile.put(ch);
12    } while (ch!='. ');
13    return 0;
14 }
```

```
C:\MinGW\bin>writeFile
Type some text (type a dot to finish):
Hola estoy escribiendo en este fichero y ahora escribo un punto y me voy .
C:\MinGW\bin>
```

Ahora, en el directorio USB:\MinG\bin apareció un fichero llamado test.txt con contenido:

```
1 Hola estoy escribiendo en este fichero y ahora escribo un punto y me voy .
```

## PROBLEM SOLVING METHODS:

- Algorithms.
- FlowCharts.

Algoritmo - Es un sistema de detallar una forma de hacer las cosas.

Collectively, these instructions are called an *algorithm*.

1. Display a message on the screen asking “How many hours did you work?”
- 

### Programs and Programming Languages

2. Wait for the user to enter the number of hours worked. Once the user enters a number, store it in memory.
3. Display a message on the screen asking “How much do you get paid per hour?”
4. Wait for the user to enter an hourly pay rate. Once the user enters a number, store it in memory.
5. Multiply the number of hours by the amount paid per hour, and store the result in memory.
6. Display a message on the screen that tells the amount of money earned. The message must include the result of the calculation performed in step 5.

```
1 //Esto es un algoritmo o PSEUDOCODIGO para calcular el salario de un trabajador cuando se
2 //introducen las horas trabajadas y el salario por hora.
3 //1. Display a message on the screen asking “How many hours did you work?”
4 //2. Wait for the user to enter the number of hours worked.
5 //3. Display a message on the screen asking “How much do you get paid per hour?”
6 //4. Wait for the user to enter an hourly pay rate.
7 //5. Multiply the number of hours by the amount paid per hour, and store the result in
8 //memory.
9 //6. Display a message on the screen that tells the amount of money earned.
```

Una vez hecho el algoritmo o PSEUDOCODIGO se puede codificar cada paso en lenguaje C++

```
1 //Esto es un algoritmo o PSEUDOCODIGO para calcular el salario de un trabajador cuando se
2 //introducen las horas trabajadas y el salario por hora.
3 # include <iostream>
4 using namespace std;
5 int main(){
6     int hours;
7     float payRate, salary;
8
9     //1. Display a message on the screen asking "How many hours did you work?
10    cout << "How many hours did you work? ";
11    //2. Wait for the user to enter the number of hours worked.
12    cin >> hours;
13    //3. Display a message on the screen asking "How much do you get paid per hour?"
14    cout << "How much do you get paid per hour? ";
15    //4. Wait for the user to enter an hourly pay rate.
16    cin >> payRate;
17    //5. Multiply the number of hours by the amount paid per hour, and store the result in
18    //memory.
19    salary = hours * payRate;
20    //6. Display a message on the screen that tells the amount of money earned.
21    cout << "Your salary is: " << salary << endl;
22    return 0;
23
24 }
```

Se compila:

```
C:\Administrator: C:\Windows\System32\cmd.exe
C:\MinGW\bin>g++ alg.cpp -o alg.exe
C:\MinGW\bin>_
```

Se ejecuta (se prueba)

```
C:\Administrator: C:\Windows\System32\cmd.exe
C:\MinGW\bin>g++ alg.cpp -o alg.exe
C:\MinGW\bin>alg
How many hours did you work? 45
How much do you get paid per hour? 7.5
Your salary is: 337.5
C:\MinGW\bin>
```

El código en realidad no necesita los comentarios. El siguiente código es el mismo pero sin comentarios:

```
1 //Esto es un algoritmo o PSEUDOCODIGO para calcular el salario de un trabajador cuando se
2 //introducen las horas trabajadas y el salario por hora.
3 # include <iostream>
4 using namespace std;
5 int main(){
6     int hours;
7     float payRate, salary;
8     cout << "How many hours did you work? ";
9     cin >> hours;
10    cout << "How much do you get paid per hour? ";
11    cin >> payRate;
12    salary = hours * payRate;
13    cout << "Your salary is: " << salary << endl;
14    return 0;
15 }
```