

## ANEXO 4.1.

---

...aclarando algunos conceptos

### **Memoria y Variables decimal, binario, hexadecimal**

Para trabajar, utilizamos un Entorno Integrado de Desarrollo (IDE), donde podemos escribir nuestro programa, compilarlo y ejecutarlo, recibir ayuda sobre errores y sintaxis, ejecutar instrucción a instrucción visualizando el valor que toman las variables.

Existen IDEs ligeros, libres y gratuitos tanto para Linux (por ejemplo, Anjuta) como para Windows (por ejemplo, DevCPP). Existen muchos otros IDEs más pesados y completos, también libres y gratuitos: Eclipse, Kdevelop, MonoDevelop, Netbeans, ... Y otros comerciales: Borland C++, Processing, Visual Studio...

### Memoria y precisión

En un microprocesador de 16 bits deseamos ejecutar un programa que ocupa 20000 posiciones de memoria y que además necesita espacio para: 10000 números enteros (16 bits), 5000 números reales de simple precisión (32 bits), 2000 números reales de doble precisión (64 bits) y 1000 alarmas binarias (1 bit). Indica la capacidad mínima de la memoria necesaria en posiciones de memoria, bits, bytes y kilobytes.

El computador debe trabajar con información de diferentes tipos: instrucciones de programas, números, letras y símbolos, etc. Las diferentes tecnologías que utiliza el computador (almacenamiento óptico en cdrom, almacenamiento magnético en discos duros, almacenamiento eléctrico en transistores, etc.) hacen que dicha información sólo se pueda almacenar utilizando secuencias de dos estados (bits), a los que llamamos estado 0 y estado 1. Por lo tanto, debemos encontrar la manera de codificar la información como secuencias de ceros y unos.

Además, en las tecnologías que almacenan o transportan la información se pueden producir errors que alteren dicha información. Para subsanar este problema, a los datos a almacenar o transportar se les añade un pequeño porcentaje de información redundante que permite detectar si se ha producido error, e incluso corregirlo. Estamos hablando de los códigos detectores y correctores de errores.

En las siguientes direcciones podemos aprender estos conceptos

(aviso: la información en Wikipedia en inglés actualmente está mucho mejor):

- [http://es.wikipedia.org/wiki/Sistema\\_de\\_numeración](http://es.wikipedia.org/wiki/Sistema_de_numeración)
- [http://es.wikipedia.org/wiki/Tipo\\_de\\_dato\\_entero](http://es.wikipedia.org/wiki/Tipo_de_dato_entero)
- [http://es.wikipedia.org/wiki/IEEE\\_punto\\_flotante](http://es.wikipedia.org/wiki/IEEE_punto_flotante)
- [http://es.wikipedia.org/wiki/Codificación\\_de\\_caracteres](http://es.wikipedia.org/wiki/Codificación_de_caracteres)
- [http://es.wikipedia.org/wiki/Detección\\_de\\_errores](http://es.wikipedia.org/wiki/Detección_de_errores)

Ej de mismo numero en codigo:

Binario  
10011101.11001

Hexadecimal  
BE.A7

Decimal  
128.75