



# UNIVERSIDAD DE GUANAJUATO, DICIS

Inteligencia Artificial

Tarea 1: Lenguaje C: Tipos de datos

Profesor: Dr. García Capulín Carlos Hugo

Martínez Lona Verónica Montserrat

Lunes 13 de Febrero, 2017

## Introducción

Verificar el tamaño en memoria de los tipos de datos en el lenguaje C, compartiendo los resultados con los compañeros de clase para hacer una comparativa entre diferentes compiladores y sistemas operativos.

## Desarrollo

El código es bastante sencillo, consiste en varios printf(), cada uno muestra en la terminal el tamaño en memoria de un tipo de dato concreto.

Código del programa implementado

```
/*Inteligencia artificial: Tarea 1, tipos de datos Lenguaje C
Verificar el tamaño en memoria de los tipos de datos en el lenguaje C,
compartiendo los resultados con los compañeros de clase para hacer una
comparativa entre diferentes compiladores y sistemas operativos.

Martínez Lona Verónica Montserrat
*/
#include<stdio.h>

int main(void)
{
    char caracter, *pchar;
    int entero, *pint;
    short int short_entero;
    long int logn_entero;
    float flotante, *pfloat;
    double doble, *pdouble;
    long double long_doble;

    printf("El tamaño en bytes del tipo char es : %i \n", sizeof(caracter));
    printf("El tamaño en bytes del tipo int es : %i \n", sizeof(entero));
    printf("El tamaño en bytes del tipo short int es : %i \n", sizeof(short_entero));
    printf("El tamaño en bytes del tipo long int es : %i \n", sizeof(logn_entero));
    printf("El tamaño en bytes del tipo float es : %i \n", sizeof(flotante));
    printf("El tamaño en bytes del tipo double es : %i \n", sizeof(doble));
    printf("El tamaño en bytes del tipo long double es : %i \n", sizeof(long_doble));

    printf("El tamaño en bytes del tipo *char es : %i \n", sizeof(pchar));
    printf("El tamaño en bytes del tipo *int es : %i \n", sizeof(pint));
    printf("El tamaño en bytes del tipo *float es : %i \n", sizeof(pfloat));
    printf("El tamaño en bytes del tipo *double es : %i \n", sizeof(pdouble));

    return 0;
}
```

## Pruebas y resultados

Debido a lo simple de la tarea, la única prueba realizada fue correrlo individualmente y tomar nota de los resultados para compartirlos con el grupo. No había datos de entrada.

```
tsukimo@MadHatter: ~/Documents/InteligenciaArtificial
tsukimo@MadHatter:~/Documents/InteligenciaArtificial$ ./T01
El tamaño en bytes del tipo char es : 1
El tamaño en bytes del tipo int es : 4
El tamaño en bytes del tipo short int es : 2
El tamaño en bytes del tipo long int es : 8
El tamaño en bytes del tipo float es : 4
El tamaño en bytes del tipo double es : 8
El tamaño en bytes del tipo long double es : 16
El tamaño en bytes del tipo *char es : 8
El tamaño en bytes del tipo *int es : 8
El tamaño en bytes del tipo *float es : 8
El tamaño en bytes del tipo *double es : 8
```

Resultados obtenidos

	char	int	Short int	Long int	float	double	Long double	*char	*int	*float	*double
Ubuntu x64	1	4	2	8	4	8	16	8	8	8	8
MacOs x64	1	4	2	8	4	8	16	8	8	8	8
Win8 x64	1	4	2	8	4	8	12	8	8	8	8

## Conclusión

El tamaño de los tipos de datos depende de diversos valores, principalmente del compilador, que es el que realiza la asignación de memoria a las variables. Otros factores que influyen en el tamaño en memoria es el sistema operativo y el hardware. Por ejemplo, en una computadora con un compilador de 32bits es altamente probable que el modificador *long* no haga cambios en un *double* y el tamaño en memoria de un *long double* sea igual que el de un *double*.

Es importante tener en cuenta estos números, pues de ellos depende la eficiencia en memoria de nuestros programas, especialmente si se utiliza mucha memoria; ya que es crítico que el espacio de la memoria no se termine inesperadamente, pues esto llevaría al fallo crítico de nuestro software.