

EXAMEN PRÁCTICAS

ARQUITECTURA DE SISTEMAS AUDIOVISUALES II

4º SAM, URJC

Fuenlabrada, 10 de Diciembre de 2013

Notas Importantes:

- En el escritorio (\$HOME/Escritorio) deberás crear un fichero en el que implementarás el programa solicitado. Deberás nombrar el fichero de la siguiente manera: **Apellido1_Apellido2_Nombre_titulación.asm**. Por ejemplo, Daniel Pérez García perteneciente a ISAM nombrará a su programa de la siguiente manera: Perez_Garcia_Daniel_ISAM.asm.
- Dentro del programa debes escribir también tu nombre y tus apellidos a modo de comentario.
- Cuando termines el examen avisa al profesor para proceder a la recogida del mismo. NO apagues el ordenador ni salgas de la sesión.

Implementa un programa ensamblador recursivo que resuelva el problema de las **Torres de Hanoi**. El juego, en su forma más tradicional, consiste en tres varillas verticales. En una de las varillas se apila un número indeterminado de discos que determinará la complejidad de la solución, por regla general se consideran ocho discos. Los discos se apilan sobre una varilla en tamaño decreciente. No hay dos discos iguales, y todos ellos están apilados de mayor a menor radio en una de las varillas, quedando las otras dos varillas vacantes. El juego consiste en pasar todos los discos de la varilla ocupada (es decir la que posee la torre) a una de las otras varillas vacantes. Para realizar este objetivo, es necesario seguir tres simples reglas:

1. Sólo se puede mover un disco cada vez.
2. Un disco de mayor tamaño no puede descansar sobre uno más pequeño que él mismo.
3. Sólo puedes desplazar el disco que se encuentre arriba en cada varilla.

Existen diversas formas de realizar la solución final, todas ellas siguiendo estrategias diversas.

A continuación se muestra un programa en C que se puede utilizar como ayuda para escribir el programa en ensamblador:

```

/* move n smallest disks from start to finish using extra */
void hanoi(int n, int start, int finish, int extra)
{
    if(n != 0)
    {
        hanoi(n-1, start, extra, finish);
        print_string("Move disk");
        print_int(n);
        print_string("from peg");
        print_int(start);
        print_string("to peg");
        print_int(finish);
        print_string("\n");
        hanoi(n-1, extra, finish, start);
    }
}

main()
{
    int n;
    print_string("Introduce el número de discos (1-8)>");
    n = read_int();
    hanoi(n, 1, 2, 3);
    return 0;
}

```