R-222 Arquitectura del Computador - LCC

Examen final condición regular

Mesa 8 de febrero de 2024

Nombre y apellido: _____

Legajo: _____

IMPORTANTE: Justificar las respuestas e indicar de manera clara y detallada los procedimientos realizados para llegar a las soluciones.

1. (25 puntos) Dado el siguiente código en Assembler x86-64:

```
.data
str:    .asciz "%x\n"
a:      .byte -55, -80
f:      .quad -56

.text
.global main
main:
        0x0000000000401126 <+0>:     push   %rbp
        0x0000000000401127 <+1>:     mov    %rsp, %rbp
        0x000000000040112a <+4>:     sub    $0x10, %rsp
        0x000000000040112e <+8>:     mov    0x404034, %al       # al =.....................
        0x0000000000401135 <+15>:    mov    %al, -0x1(%rbp)
        0x0000000000401138 <+18>:    mov    0x404035, %al       # al =.....................
        0x000000000040113f <+25>:    mov    %al, -0x2(%rbp)
        0x0000000000401142 <+28>:    add    -0x1(%rbp), %al     # al =.....................
        0x0000000000401145 <+31>:    mov    0x404036, %rax      # rax =....................
        0x000000000040114d <+39>:    mov    %rax, -0x10(%rbp)
        0x0000000000401151 <+43>:    lea    0x404030, %rdi      # rdi =....................
        0x0000000000401159 <+51>:    mov    -0x10(%rbp), %esi   # esi =....................
        0x000000000040115c <+54>:    call   printf
        0x0000000000401161 <+59>:    add    $0x10, %rsp
        0x0000000000401165 <+63>:    mov    %rbp, %rsp
        0x0000000000401168 <+66>:    pop    %rbp
        0x0000000000401169 <+67>:    ret
```

   (a) Realizar un esquema para mostrar la ubicación en memoria de las variables. Suponer que la etiqueta `str` corresponde a la dirección de memoria `0x404030`.

   (b) Indicar el valor de los registros indicados en los comentarios luego de ejecutarse la instrucción correspondiente.

   (c) Realizar un esquema mostrando el estado de la pila antes de ejecutarse el epílogo. Suponer como valores iniciales `rsp=0x7fffffffebc8` y `rbp=0x1`.

   (d) Indicar lo que se imprime al ejecutarse la instrucción `call printf`.

   (e) Indicar el valor de retorno luego de ejecutarse el código.

   (f) Indicar de las banderas `CF` y `OF` luego de ejecutarse la instrucción `addb -1(%rbp), %al`

2. (25 puntos) Dado el siguiente código en C:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[]) {
    if (argc < 2){
        printf("La cantidad de argumentos es insuficiente\n");
    } else {
        int suma=0;
        for (int i=1; i<argc; i++){
            suma = suma + atoi(argv[i]);
        }
        printf("Cantidad de argumentos: %d, la suma de los argumentos es: %d\n", argc, suma);
    }
    return 0;
}
```

Escribir un código equivalente en Assembler x86-64.

Nota: La función `atoi` de la librería `stdlib.h` convierte una cadena de caracteres a su valor numérico equivalente (entero). Toma como parámetro de entrada la cadena a convertir y devuelve el valor numérico correspondiente.

3. (25 puntos) Dado el siguiente código en C:

```
#include <stdio.h>

float a[5] = {1.0, 2.0, 3.0, 4.0, 5.0};
float b[5] = {2.0, 4.0, 6.0, 8.0, 10.0};
float c[5];

int main()
{
    float suma=0;
    for(int i=0; i<5; i++){
        printf("%f\n", c[i] = a[i] + b[i]);
        suma = suma + c[i];
    }
    printf("La suma de las componentes es: %f\n", suma);
    return 0;
}
```

Escribir un código equivalente en Assembler ARM.

4. (25 puntos) Se tiene un sistema de memoria virtual con paginación de dos niveles con 10 bits en cada nivel, que puede direccionar un total de 2^{32} bytes y tiene 1 GB de memoria principal con páginas virtuales y físicas de 4 KB:
   (a) ¿Cuántos bits tiene la dirección virtual?
   (b) Indicar cómo está compuesta la dirección virtual.
   (c) ¿Cuántos bits tiene la dirección física?
   (d) ¿Cuál es el número máximo de páginas virtuales en el sistema?
   (e) ¿Cuál es el número máximo de páginas físicas en el sistema?
   (f) Atendiendo al contenido de la TLB y de las tablas de páginas de un proceso mostradas a continuación, indicar la dirección de memoria accedida para las siguientes direcciones virtuales:
      1. 0xff8021f0
      2. 0x8031ff
      3. 0x1003ff0

TLB

V	Nro. Pág. Virtual	Nro. Pág. Física
0		
1	0x1c00	0x600
0		
1	0x1003	0x5ff
0		

Tabla de páginas de primer nivel

	V	Dirección
0x3ff	0	
	1	0x1400
	1	0x1c00
	0	
	1	0x2000
	1	0x1000
	:	:
	0	
	0	
	0	
	1	0x1800
0x0	0	

Tablas de páginas de segundo nivel

	V	Nro. Página
0x1bff	0	
	0	
	1	0x2ff
	:	:
	1	0x2f0
	0	
	0	
0x1800	1	0x2fa

	V	Nro. Página
0x17ff	0	
	1	0x1df
	0	
	:	:
	0	
	1	0x1d0
	0	
0x1400	0	

	V	Nro. Página
0x13ff	0	
	1	0x5f0
	0	
	:	:
	0	
	0	
	1	0x600
0x1000	0	

Ejercicio	1	2	3	4	Total
Puntos	25	25	25	25	100
Puntos obtenidos					