**Estructura de los computadores**

Practica 7 8 9

Francisco Joaquín Murcia Gómez 48734281H

Grupo 3

**Índice**

**Practica 4**………………………………………………………………………3-10

Aritmética de enteros (3) y funciones

Ejemplos …………………………………………………………………3-8

Entregas ………………………………………………………………...9-10

**Practica 5**…………………………………………………………….………11-14

Estructuras de control

Ejemplos ………………………………………...……………………11-12

Entregas …………………………………………………………...….13-14

**Practica 6**….…………………………………………………………………15-18

Variables

Ejemplos …………………………………………………………………...15

Entregas ………………………………………………………………16-18

**Practica 7**

**Ejemplos**

1. **Vectores de caracteres**
   1. **¿Cuántos caracteres tiene la cadena?**

30

* 1. **¿En qué dirección de la memoria se encuentra el carácter null)?**

1e (1c+4)

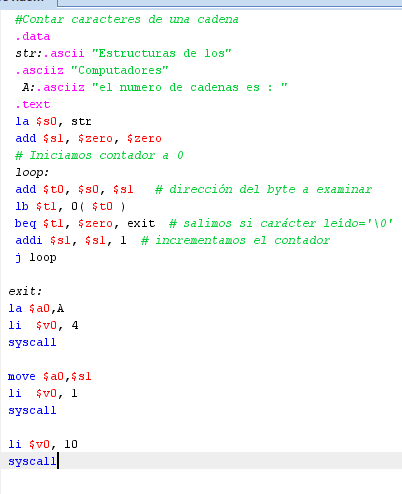
* 1. **¿Cómo se actualiza el índice del vector?**

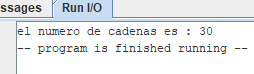
$S1

* 1. **¿El programa funcionaría si la cadena solo constara del carácter null?**

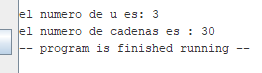
Si, pero terminaría inmediatamente

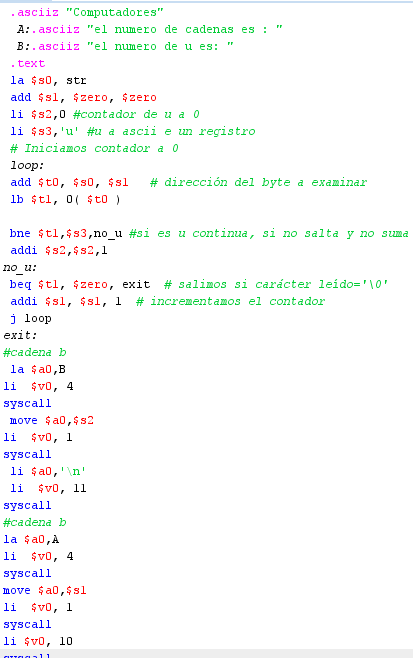
* 1. **Modifica el código para que muestre por pantalla el mensaje “El número de caracteres de la cadena es: “ y a continuación el resultado.**





* 1. **Modifica el código para que calcule el número de veces que se repite la vocal “u”**





1. **Vectores de enteros**
   1. **Ensambla y ejecuta el programa. Comprueba que el vector A se copia en el vector B. ¿En qué dirección empieza el vector B?**

0x10010014

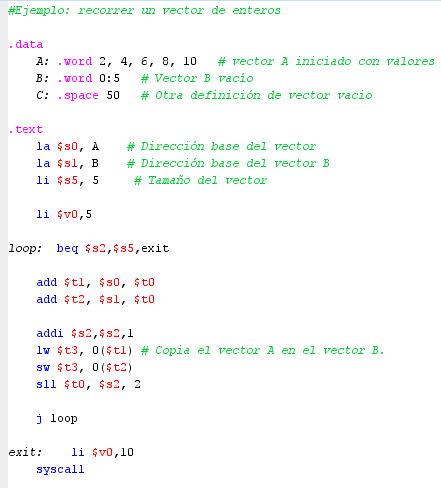
* 1. **¿Porque no se pueden acabar los vectores con el carácter null igual que se hace con las cadenas de caracteres?**

Porque no son caracteres, son enteros

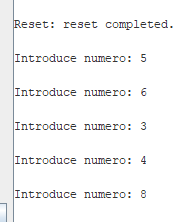
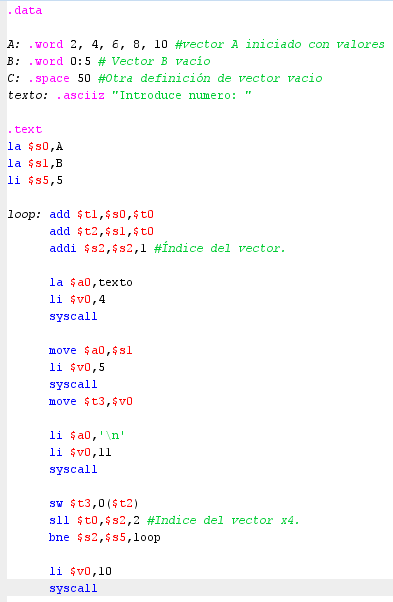
* 1. **En el programa se recorren los vectores actualizando el índice con la instrucción sll. ¿De qué otra manera se podrían recorrer los vectores?**

Con un acumulador de 4

* 1. **Se ha utilizado un bucle del tipo do-while, modifica el programa por que el bucle sea de tipo for-while.**



* 1. **Modifica el programa para que el vector B se rellene con enteros leídos del teclado. Previamente se tiene que mostrar un mensaje por consola que pida los elementos**



1. **Direccionamiento en memoria**
   1. **¿Cuántas pseudoinstrucciones contiene el código?**

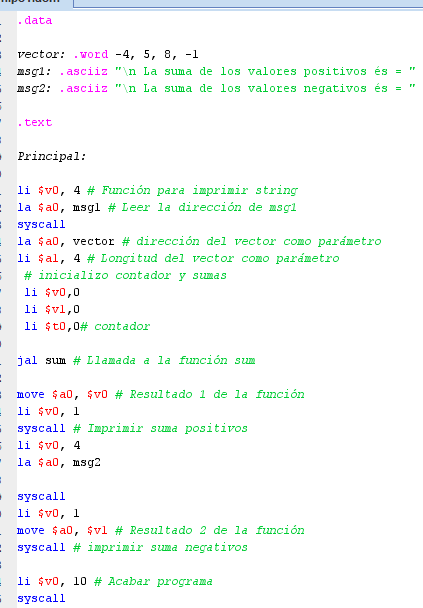
2, “la” y “sw”

* 1. **Ensambla el código y observa la traducción de las pseudoinstrucciones en instrucciones del MIPS. ¿En qué instrucciones se ha traducido sw $t3, C? ¿Qué registro auxiliar se ha utilizado?**

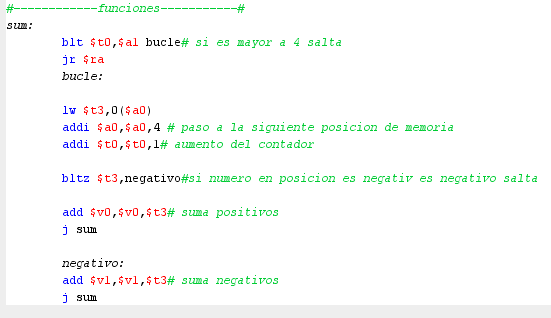
En “lui” y “sw”, $1

**Entregas**

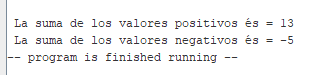
**Haz el código de la función sum que calcula la suma de los valores positivos y negativos del vector, dirección del cual se pasa como parámetro en $a0 y la longitud en $a1. La función devuelve en $v0 la suma de los valores positivos y en $v1 la suma de los negativos. Recuerda que en la función tienes que utilizar los registros $tj.**



Funciones:



Consola:

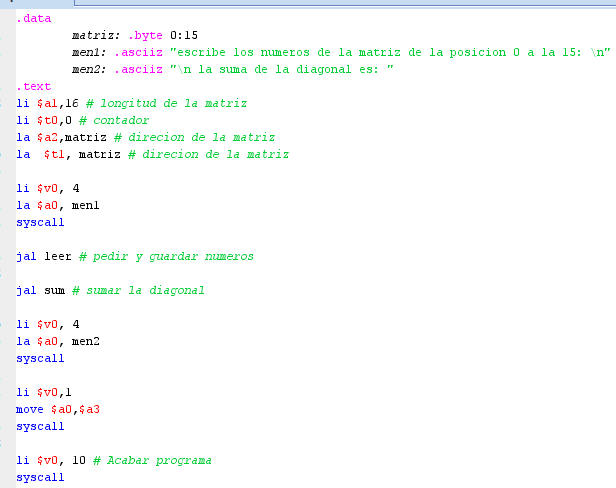


Ventana de registros $v0,$v1

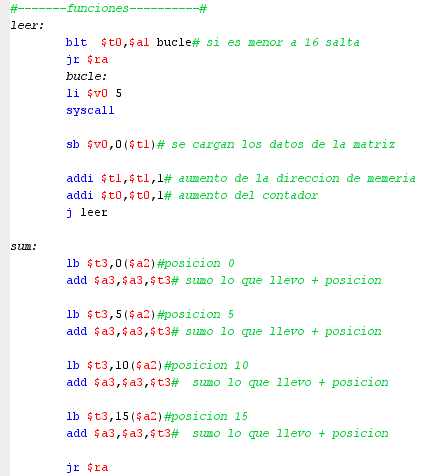


**Haz el código que calcula la suma de los elementos de la diagonal principal de una matriz 4x4 de valores enteros introducida por teclado. Muestra la suma por pantalla.**

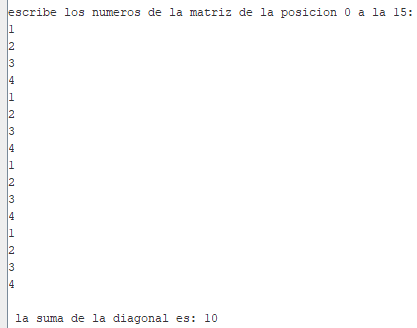
main



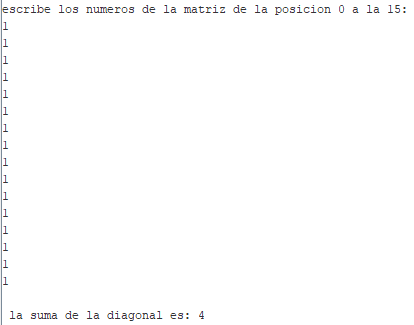
Funciones:



Funcionamiento:

Introducimos la matriz A=[1,2,3,4; 1,2,3,4; 1,2,3,4; 1,2,3,4]

Introducimos la matriz B=[1,1,1,1; 1,1,1,1; 1,1,1,1; 1,1,1,1;]



**Practica 8**

**Ejemplos**

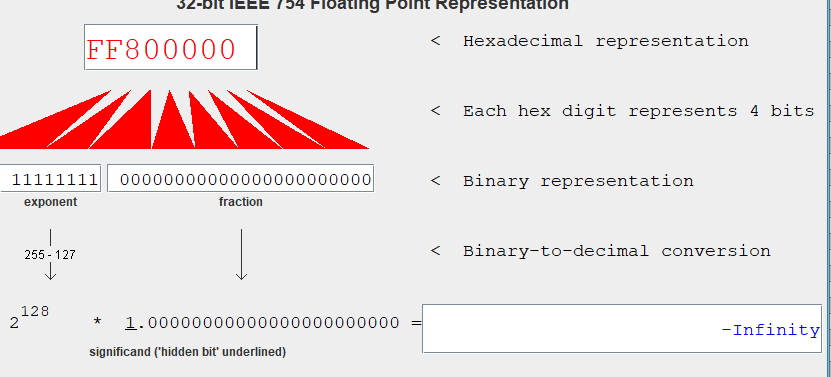
1. **Operaciones aritméticas**
   1. **¿Cuál es la razón por la que no hay instrucciones aritméticas en coma flotante con datos inmediatos?**

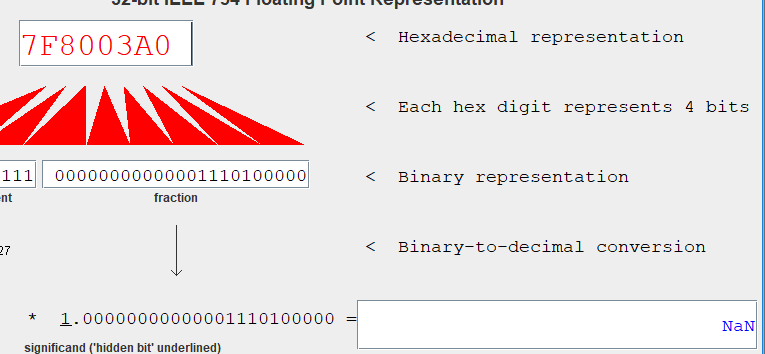
Porque hay que pasarlo a IEEE

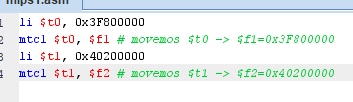
* 1. **¿Por qué no hay registros HI y LO para guardar el resultado de la multiplicación y división en coma flotante del mismo modo que con los números enteros?**

Porque te lo da en IEEE

1. **Operaciones de movimiento de datos**
   1. **Haciendo uso de la herramienta de representación en coma flotante del MARS, comprueba que realmente en $f12 hay un -∞ y en $f20 un NaN según el estándar IEEE 754.**





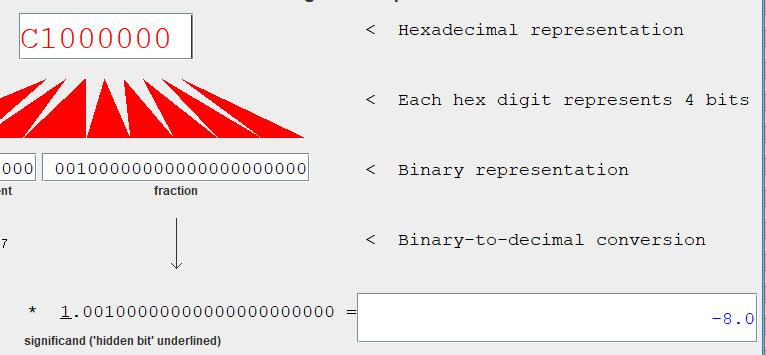
* 1. **Haz que el contenido de $f1 sea el valor 1 en coma flotante y el de $f2 el valor - 2.5 en coma flotante.**



* 1. **Di una manera de escribir un 0.0 en $f0 con sólo una instrucción máquina.**

mtc1 $zero,$f0

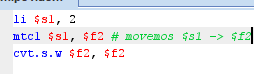
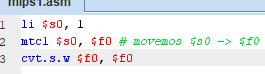
1. **Conversión de tipo**
   1. **Haciendo uso de la herramienta de representación en coma flotante del MARS, comprueba que realmente en $f0 después de la conversión hay un -8.**



* 1. **¿Qué valor consideraría la máquina que habría en $f0 si no hiciésemos la conversión con la instrucción cvt.s.w?. Aprovecha que puedes ver contenidos en decimal y en hexadecimal.**

Un NaN

* 1. **Haz que el contenido de $f1 sea el valor 1 en coma flotante y el de $f2 el valor -2 en coma flotante utilizando las instrucciones de conversión de tipo.**



* 1. **¿Cuál es la razón por la que al finalizar el programa los contenidos de $s0 y $s1 son distintos?**

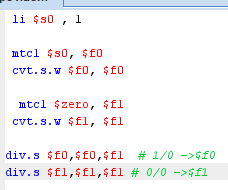
Por el redondeo

* 1. **Ensambla y ejecuta el código. ¿Qué valor representa en el formato IEEE 754 el contenido final de $f0?**

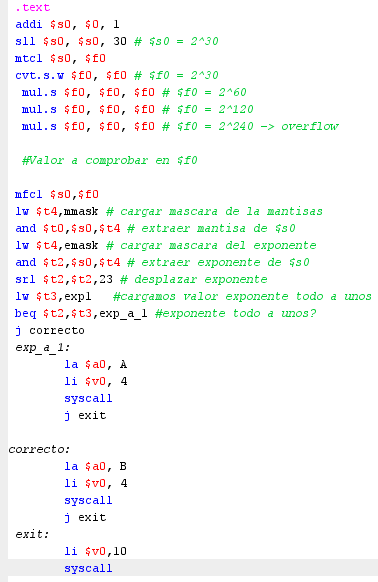
Infinito

* 1. **Observa que no ha ocurrido ninguna excepción a la ejecutar el código. ¿Ocurre lo mismo al dividir por cero? ¿Y al hacer 0/0? Añade instrucciones al código anterior y haz la prueba.**



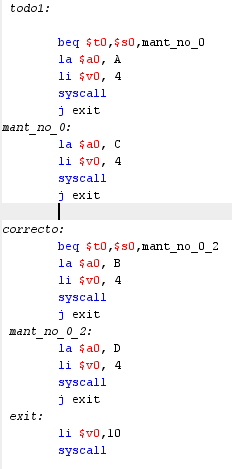
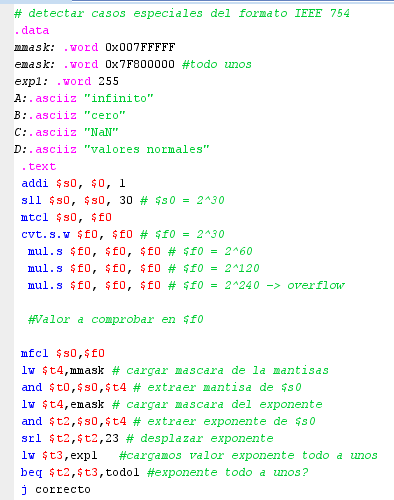


* 1. **Completa el código para que muestre en consola un mensaje de error por desbordamiento. Comprueba que funciona correctamente**





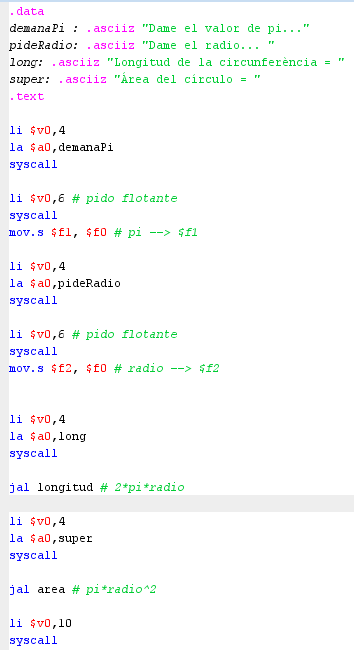
* 1. **Completa el código para que detecte todos los casos especiales y muestre mensajes en la consola. Haz distintas pruebas y comprueba que funciona.**



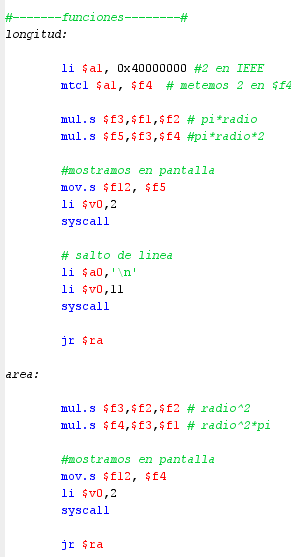
**Entregas**

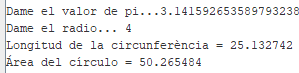
**Completa el siguiente código de partida que pide el radio por teclado y tiene que calcular y mostrar en la consola la longitud de la circunferencia y el área del círculo.**

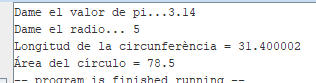
El main del programa



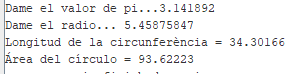
Funciones del programa:



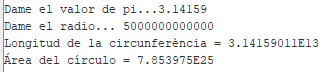
Pruebas:



El radio puede ser tambien en coma flotante

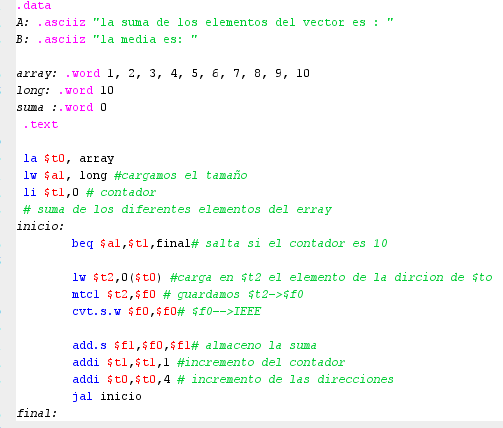


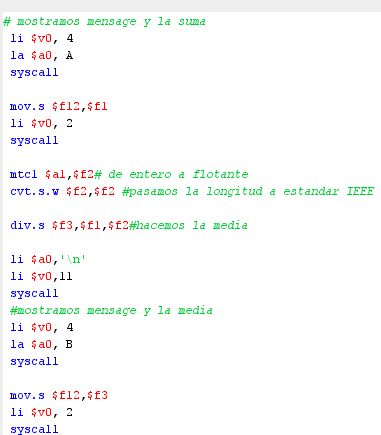
Si el radio es muy grande el resultado te lo pone en notacion cientifica



**A partir de la siguiente declaración de un vector de 10 elementos:**

**Haz el código que suma los elementos del vector y calcula el valor medio. Muestra el resultado por la consola.**







Registros:

Carga el 1 de la1ª posición y lo añade a la suma,



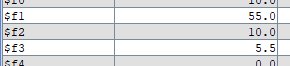
Carga el 3 de la3ª posición y lo añade a la suma,



Carga el 8 de la 3ª posición del array y lo añade a la suma



$f1 la suma,$f2 la longitud y $ f3 la media



**Practica 9**

**Ejemplos**

1. **Operaciones de movimiento de datos**
   1. **¿Cuál es la razón por la que el registro base de las instrucciones lwc1 y swc1 pertenecen al banco de registros de enteros y no de la FPU?**

Porque las direcciones son enteros

* 1. **Ensambla, ejecuta el programa y observa el contenido que adquieren los registros para verificar los resultados que has obtenido a mano. ¿Qué conclusión puedes sacar?**

Que ha habido un redondeo en $f4

1. **Instrucciones de comparación**
   1. **No hay comparación mayor que, ¿cómo lo podéis solucionar?**

Intercanvio de posiciones

1. **Saltos condicionales para coma flotante**
   1. **¿Dónde crees que se ejecutará la instrucción bclt? ¿En la CPU o en la FPU?**

En la cpu,los flag son enteros

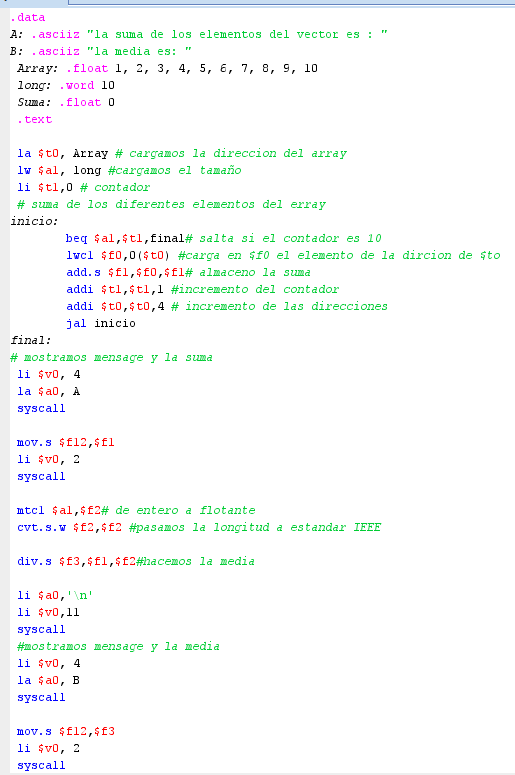
* 1. **¿Qué código de condición se ve afectado? ¿Hasta cuándo permanecerá el valor del código de condición?**

El 1 ; hasta que haya una comparación

**Entregas**

**A partir de la siguiente declaración de un vector de 10 elementos:**

**Haz el código que suma los elementos del vector y calcula el valor medio. Muestra el resultado por la consola.**





Registros:

Carga el 1 de la1ª posición y lo añade a la suma,



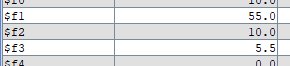
Carga el 3 de la3ª posición y lo añade a la suma,



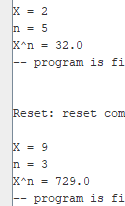
Carga el 8 de la 3ª posición del array y lo añade a la suma



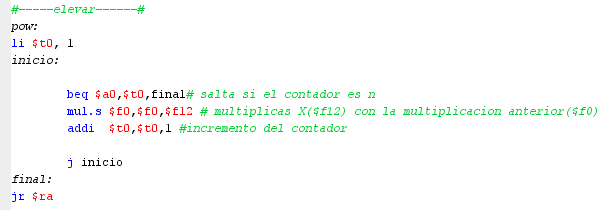
$f1 la suma,$f2 la longitud y $ f3 la media



**Implementar la función float pow(float x;int n) que calcula la potencia n-ésima de x. Los argumentos y los valores se pasan según convenio: x en $f12, n en $a0. El resultado se devuelve en $f0. Utilizad el siguiente código de partida:**



Funcion pow a completar:



**Implementar la función max que nos devuelve el valor mayor de dos números en coma flotante. Los argumentos se pasan según convenio en $f12 y $f14 y el resultado se devuelve en $f0. Utilizad el siguiente código de partida:**

**funcion**

