

Segundo examen parcial

Laboratorio de Simulación

Luis Alfredo Ixquiac Méndez
carnet: 201704290

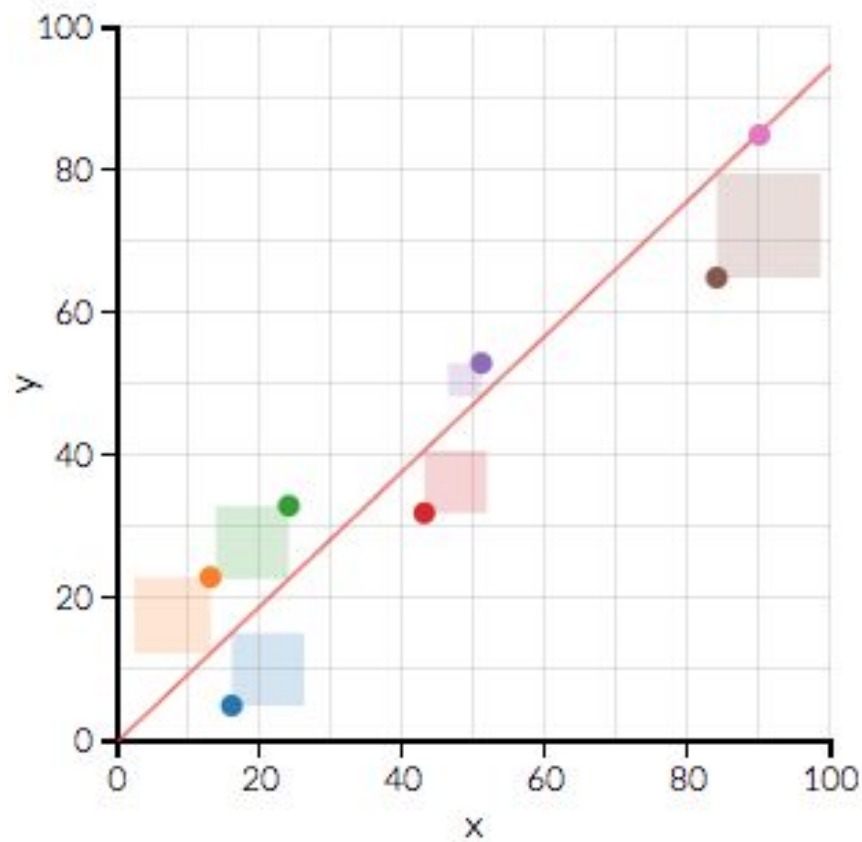
Mínimos cuadrados: Evolución atlética

Un atleta necesita pasar la evaluación física para poder ingresar a la selección de su equipo. Como mínimo debe de poder de realizar diez sentadillas con 150 Kg de peso. 3 meses antes de la evaluación sufre un desgarre en el cuádriceps, derecho lo que le permitió obtener la siguiente evolución.

semana	peso (Kg)
1	20
2	26
3	31
4	38
5	45
6	49
7	54

Le solicitan que genere un programa el cual cumpla con las siguientes condiciones:

- Una gráfica que compare los valores tabulados y la recta que mejor aproxima el crecimiento.
- Estime el el peso que logra levantar el atleta después de 3 meses, este logra pasar la prueba para ingresar al equipo.



$$m = \frac{n \sum_{k=1}^n (x_k y_k) - \sum_{k=1}^n x_k * \sum_{k=1}^n y_k}{n \sum_{k=1}^n x_k^2 - (\sum_{k=1}^n x_k)^2}$$

$$b = \frac{\sum_{k=1}^n y_k - m \sum_{k=1}^n x_k}{n}$$

$$\Delta m = \frac{\sqrt{n} \epsilon}{\sqrt{n \sum_{k=1}^n x_k^2 - (\sum_{k=1}^n x_k)^2}}$$

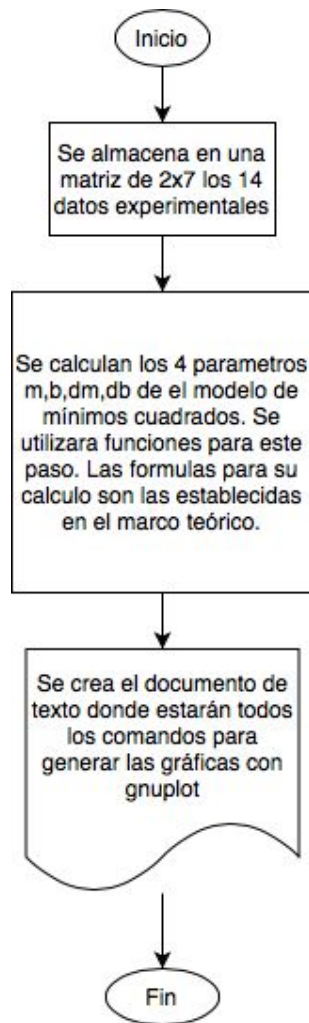
$$\Delta b = \frac{\epsilon}{\sqrt{n}}$$

semana	peso (Kg)
1	20
2	26
3	31
4	38
5	45
6	49
7	54

```
#!/bin/bash

echo "INICIANDO PROGRAMA PARA ENCONTRAR LA REGRESION LINEAL DE LOS DATOS DE EVOLUCION ATLETICA"
gcc minimos.c -lm
./a.out
echo "INICIANDO PROCESO PARA GENERAR LA GRAFICA"
gnuplot graficador.txt
echo "El archivo grafica.jpeg se a creado"
echo "FIN DE EL PROGRAMA"
```

Semana	Peso
1	20
2	26
3	31
4	38
5	45
6	49
7	54



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

//Ingresamos los datos experimentales
float dat[2][7]={{1,2,3,4,5,6,7},{20,26,31,38,45,49,54}};

//Definimos los prototipos de las funciones a utilizar
//t representa a X o a Y: t0=x t1=y
float st(int i);
float st2(int i);
float sxy();
float m(float,float,float,float,float);
float b(float,float,float,float);
float dm(float,float,float,float);
float db(float,float);
float r(float,float,float,float,float,float);
```

```
float st(int i){
    float res=0;
    for(int j=0;j<7;j++){
        res=res+dat[i][j];
    }
    return res;
}

float sxy(){
    float res=0;
    for(int j=0;j<7;j++){
        res=res+(dat[0][j])*(dat[1][j]);
    }
    return res;
}

float st2(int i){
    float res=0;
    for(int j=0;j<7;j++){
        res=res+(dat[i][j])*(dat[i][j]);
    }
    return res;
}
```

```
float m(float n, float sx, float sy, float sxy, float sx2){  
    return (((n)*(sxy))-((sx)*(sy)))/(((n)*(sx2))-((sx)*(sx)));  
}  
  
float b(float n, float sx, float sy, float m){  
    return (((sy)-((m)*(sx)))/(n));  
}  
  
float dm(float n, float sx, float sx2, float e){  
    return (((sqrt(n))*(e))/(sqrt(((n)*(sx2))-((sx)*(sx)))));  
}  
  
float db(float n, float e){  
    return ((e)/(sqrt(n)));  
}
```



```

void main(void){
    printf("\n***INICIANDO PROGRAMA EN C***\n\n");

    float n=7; //numero de parejas de datos experimentales
    float e=0.2; //error en la variable x de los datos experimentales
    float vm=m(n,st(0),st(1),sxy(),st2(0)); //Valor de la pendiente
    float vb=b(n,st(0),st(1),vm); //Valor del intersepto con el eje y
    float vdm=dm(n,st(0),st2(0),e); //Incertesa en la pendiente
    float vdb=db(n,e); //Incertesa en el intersepto con el eje y

    printf("REGRECIION LINEAL COMPLETADA\n");
    printf("Los valores obtenidos son:\n\n");
    printf("El valor de m es: m=%f\n",vm);
    printf("El valor de b es: b=%f\n",vb);
    printf("El valor de dm es: dm=%f\n",vdm);
    printf("El valor de db es: db=%f\n",vdb);
}

```

```
printf("GENERANDO ARCHIVO PARA GRAFICAR");

char lineal[100];
sprintf(lineal,"plot ((%f)*x)+(%f),'data.txt' using 1:2 with points pt 4 ps 2 \n",vm,vb);

FILE* fichero;
fichero = fopen("graficador.txt", "wt");
fputs("set terminal jpeg\n", fichero);
fputs("set title 'Evolucion atletica'\n", fichero);
fputs("set xlabel 'Semana'\n", fichero);
fputs("set ylabel 'Peso(kg)'\n", fichero);
fputs("set xrange [0:11]\n", fichero);
fputs("set yrange [10:80]\n", fichero);
fputs("set grid\n", fichero);
fputs("set output 'grafica.jpeg'\n", fichero);
fputs(lineal, fichero);

fclose(fichero);

printf("\nARCHIVO PARA GRAFICAR GENERADO\n");
```

```
float est=((vm)*(12)+(vb));  
printf("\n\n\nSegun el modelo de minimos cuadrados, para la semana de la competicion, el atlet  
a podra levantar un peso de %f Kg. Esto es menor a 150Kg, por lo que el atleta no podra estar listo pa  
ra evaluacion, suponiendo que su evlucion obedezca la prediccion de este modelo\n\n\n",est);  
  
printf("\n\nPROCESO EN LANGUAGE C COMPLETADO\n");
```

```
#!/bin/bash
```

```
echo "INICIANDO PROGRAMA PARA ENCONTRAR LA REGRECION LINEAL DE LOS DATOS DE EVOLUCION ATLETICA"
```

```
gcc minimos.c -lm
```

```
./a.out
```

```
echo "INICIANDO PROCESO PARA GENERAR LA GRAFICA"
```

```
gnuplot graficador.txt
```

```
echo "El archivo grafica.jpeg se a creado"
```

```
echo "FIN DE EL PROGRAMA"
```

```
set terminal jpeg
set title 'Evolucion atletica'
set xlabel 'Semana'
set ylabel 'Peso(kg)'
set xrange [0:11]
set yrange [10:80]
set grid
set output 'grafica.jpeg'
plot ((5.785714)*x)+(14.428572), 'data.txt' using 1:2 with points pt 4 ps 2
```

```
luis@Lubuntu:~/Escritorio/examen2/problema1$ ./problema1.sh
INICIANDO PROGRAMA PARA ENCONTRAR LA REGRECIION LINEAL DE LOS
DATOS DE EVOLUCION ATLETICA
```

```
***INICIANDO PROGRAMA EN C***
```

```
REGRECIION LINEAL COMPLETADA
```

```
Los valores obtenidos son:
```

```
El valor de m es: m=5.785714
```

```
El valor de b es: b=14.428572
```

```
El valor de dm es: dm=0.037796
```

```
El valor de db es: db=0.075593
```

```
GENERANDO ARCHIVO PARA GRAFICAR
```

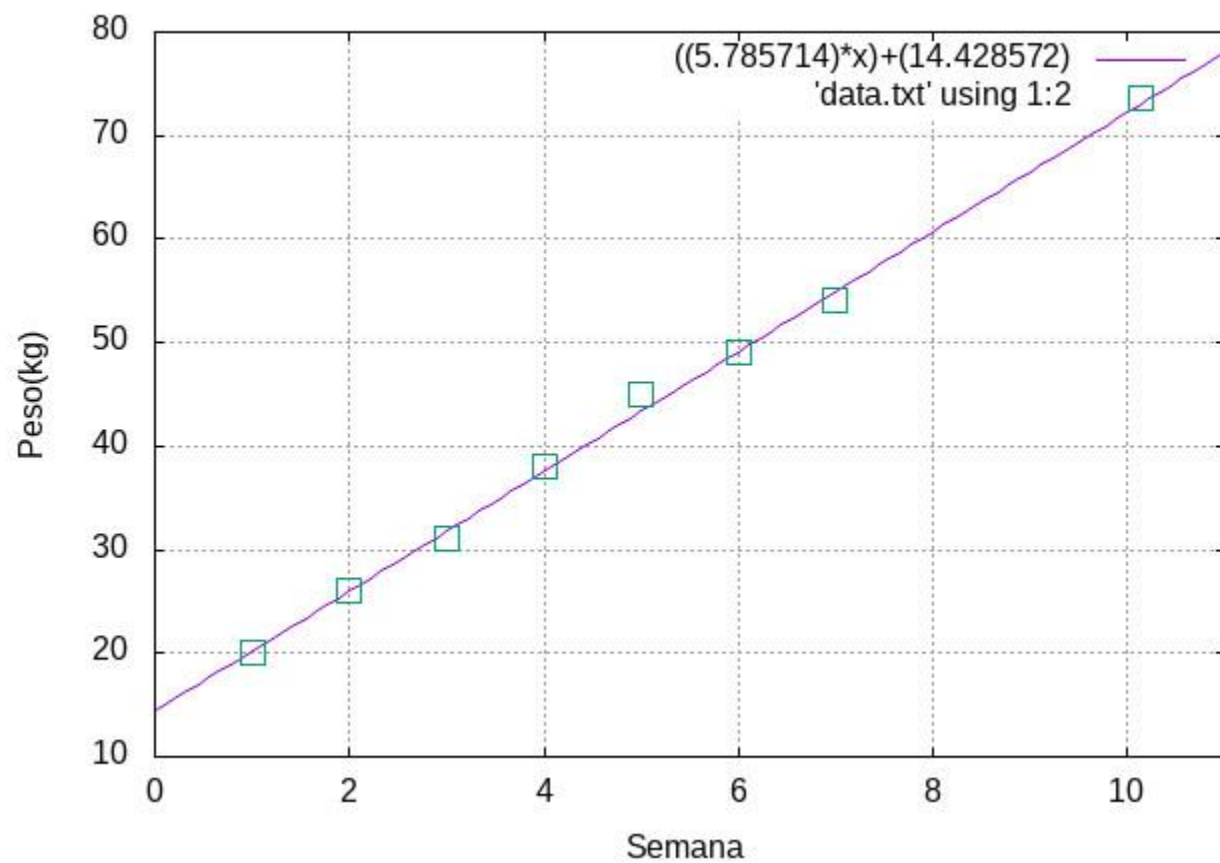
```
ARCHIVO PARA GRAFICAR GENERADO
```

Segun el modelo de minimos cuadrados, para la semana de la co
mpeticion, el atleta podra levantar un peso de 83.857147 Kg.
Esto es menor a 150Kg, por lo que el atleta no podra estar li
sto para evaluacion, suponiendo que su evlucion obedezca la p
rediccion de este modelo

```
PROCESO EN LENGUAGE C COMPLETADO
```

```
INICIANDO PROCESO PARA GENERAR LA GRAFICA
```

Evolucion atletica



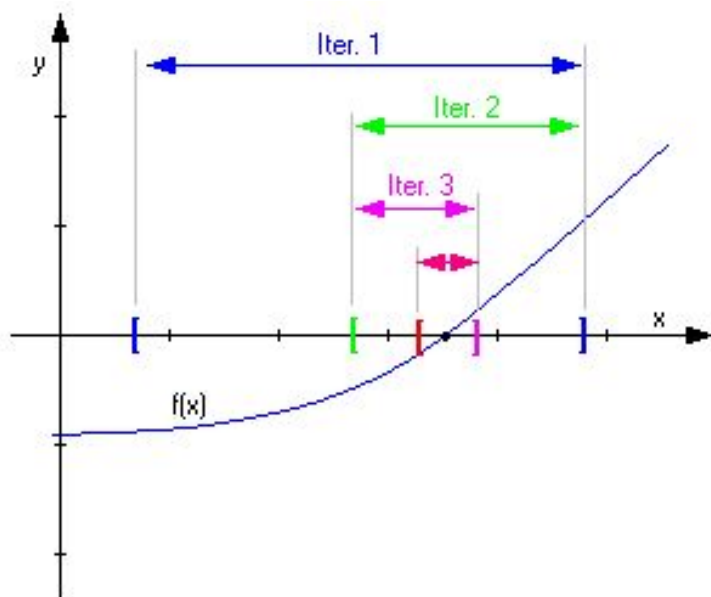
Bisección

Utilizando el método de bisección, encuentre una raíz de la ecuación

$$f(x) = \arcsin x$$

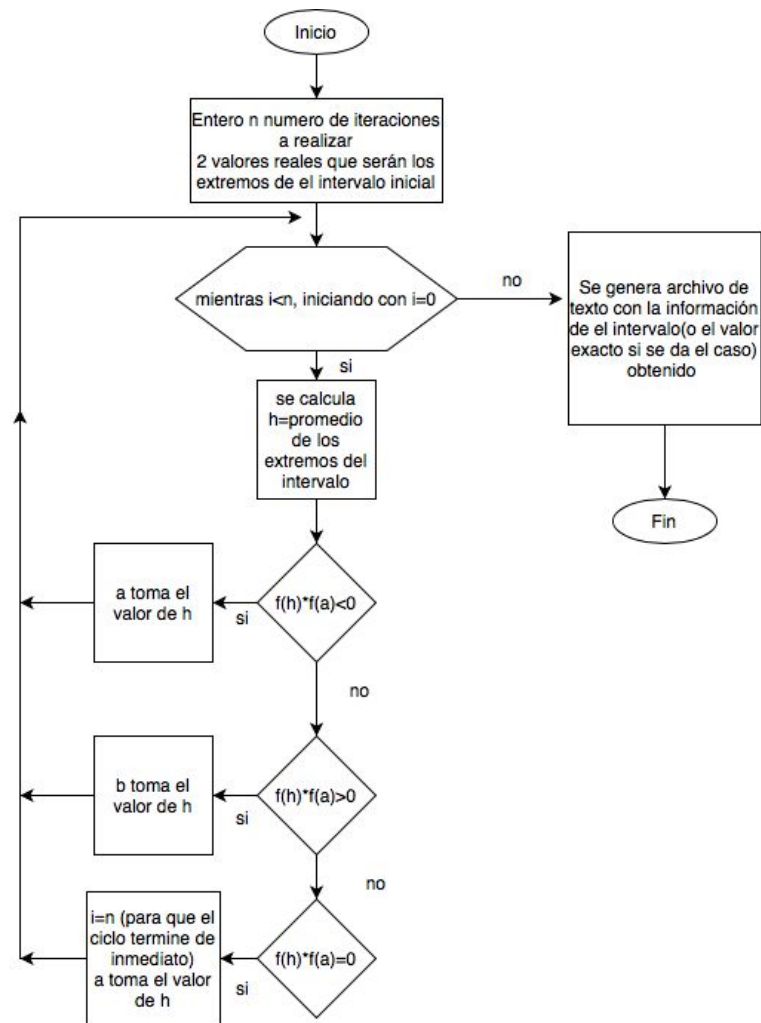
Debe de realizar la gráfica de la ecuación y comparar el resultado obtenido con el programa realizado en C.

Si f es continua en $[a,b]$ y $f(a) < 0 < f(b)$, entonces existe un x en $[a,b]$ tal que $f(x)=0$.



```
#!/bin/bash
echo "***INICIANDO PROGRAMA PARA ENCONTRAR UNA RAIZ POR MEDIO DE EL METODO DE LA BISECCION:***"
gcc biseccion.c -lm
./a.out

echo "Iniciando proceso en GNUPLOT"
gnuplot "graficador.txt"
echo "Proceso en GNUPLOT completado."
echo "La imagen grafica.jpeg a sido creada"
rm datos.txt
```



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

float pro(float, float);
void bisec(float, float, int, float*, float*, int*, int*);
```

```
float pro(float a, float b){
    return ((a+b)/2);
}

void bisec(float a, float b, int n, float *ma, float *mb, int *mi, int *flag){

    if((asin(a))*(asin(pro(a,b)))<0){
        *mb=pro(a,b);
    }else if(((asin(a))*(asin(pro(a,b))))>0){
        *ma=pro(a,b);
    }else if(((asin(a))*(asin(pro(a,b))))==0){
        *ma=pro(a,b);
        *mi=n;
        *flag=1;
    }
}
```

```
void main(void){  
  
    int n;  
    float a=-0.55, b=0.5;  
    int flag=0;  
  
    printf("\n***Calculo de una raiz de la funcion arcsin(x) mediante el metodo de la biseccion:***\n");  
    printf("\nIngresa el numero de iteraciones que desear efectuar: n= ");  
    scanf("%i",&n);  
  
    for(int i=0;i<n;i++){  
        bisec(a,b,n,&a,&b,&i,&flag);  
    }  
}
```

```
if(flag==0){
    printf("\n\n\nLa raiz se encuentra en el rango de [%f:%f]\n",a,b);
    char linea2[30];
    char linea3[30];
    float t=0;
    sprintf(linea2,"%f,%f\n",a,t);
    sprintf(linea3,"%f,%f\n",b,t);
    FILE* fichero;
    fichero = fopen("datos.txt", "wt");
    fputs("ejex,ejey\n", fichero);
    fputs(linea2, fichero);
    fputs(linea3, fichero);
    fclose(fichero);
    printf("\nPROCESO EN LANGUAGE C COMPLETADO\n");
}else if(flag==1){
    printf("La raiz es: x= %f",a);
    char linea2[30];
    float t=0;
    sprintf(linea2,"%f,%f\n",a,t);
    FILE* fichero;
    fichero = fopen("datos.txt", "wt");
    fputs("ejex,ejey\n", fichero);
    fputs(linea2, fichero);
    fclose(fichero);
    printf("\nPROCESO EN LANGUAGE C COMPLETADO\n");
}
```

```
set terminal jpeg
set datafile separator ","
set title "Metodo de la Biseccion"
set xlabel "x,root"
set ylabel "arcsin(x)"
set xrange [-1:1]
set yrange [-1.6:1.6]
set grid
set output "grafica.jpeg"
plot asin(x) ,"datos.txt" using 1:2 with linespoints pt 1 ps 3
```

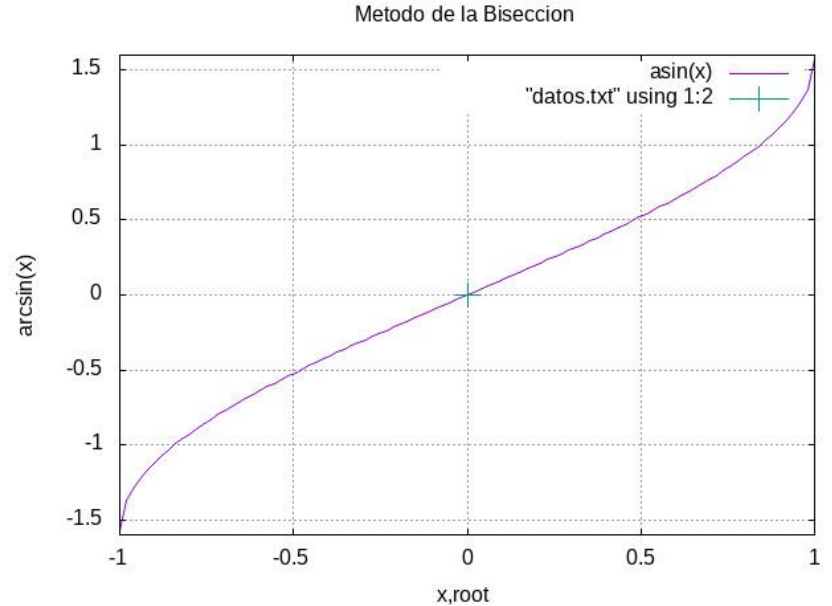
```

luis@Lubuntu:~/Escritorio/examen2/problema2$ ./problema2.sh
***INICIANDO PROGRAMA PARA ENCONTRAR UNA RAIZ POR MEDIO DE EL
METODO DE LA BISECCION:***

***Calculo de una raiz de la funcion arcsin(x) mediante el me
todo de la biseccion:***

Ingrese el numero de iteraciones que desear efectuar: n= 10
La raiz es: x= 0.000000
PROCESO EN LENGUAGE C COMPLETADO
Iniciando proceso en GNULPOT
Proceso en GNUPLOT completado.
La imagen grafica.jpeg a sido creada
luis@Lubuntu:~/Escritorio/examen2/problema2$ 

```




```

luis@Lubuntu:~/Escritorio/examen2/problema2$ ./problema2.sh
***INICIANDO PROGRAMA PARA ENCONTRAR UNA RAIZ POR MEDIO DE EL
METODO DE LA BISECCION:***

***Calculo de una raiz de la funcion arcsin(x) mediante el me
todo de la biseccion:***

Ingrese el numero de iteraciones que desear efectuar: n= 5

La raiz se encuentra en el rango de [-0.025000:0.007812]

PROCESO EN LENGUAGE C COMPLETADO
Iniciando proceso en GNULPOT
Proceso en GNUPLOT completado.
La imagen grafica.jpeg a sido creada
luis@Lubuntu:~/Escritorio/examen2/problema2$ █

```

