

## Laboratorio 1a

Nombre del archivo fuente: lab01a.c

Tiempo límite: 1

Escribir un programa en C que lea dos números enteros y escriba la suma

### Input

La entrada se debe tomar desde la entrada estandar. La entrada está compuesta por una secuencia de pares de líneas, donde cada línea del par contiene un sumando. Como primera línea se encuentra un número que indica cuantas sumas deben realizarse.

### Output

La salida debe enviarse a la salida estandar.

La salida está compuesta por una secuencia de líneas, donde en cada línea, está la suma de los operandos leídos en cada caso.

Ejemplo de entrada	Ejemplo de salida
3 10 -25 11 121 2 -2	-15 132 0

Utilizar el tipo *int* de C

## Laboratorio 1b

Nombre del archivo fuente: lab01b.c

Tiempo límite: 1

Escribir un programa en C que lea dos números reales y escriba el máximo

### Input

La entrada se debe tomar desde la entrada estandar. La entrada está compuesta por una secuencia de pares de líneas, donde cada línea del par contiene un real. Como primera línea se encuentra un número que indica cuantas comparaciones deben realizarse.

### Output

La salida debe enviarse a la salida estandar.

La salida está compuesta por una secuencia de líneas, donde en cada línea, está máximo de los reales leídos en cada caso.

Ejemplo de entrada	Ejemplo de salida
3 10 -2.5 1.1 121 2.3 -2.3	10.000000 121.000000 2.300000

Utilizar el tipo *float* de C

## Laboratorio 1c

Nombre del archivo fuente: lab01c.c

Tiempo límite: 1

Escribir un programa en C que lea un vector de números reales y escriba el mínimo del vector.

### Input

La entrada se debe tomar desde la entrada estandar. La entrada está compuesta por una secuencia de líneas, donde la primera línea contiene la cantidad de reales a leer y las siguientes líneas contienen un real. Como primera línea se encuentra un número que indica para cuantos vectores se debe encontrar el minimo.

### Output

La salida debe enviarse a la salida estandar.

La salida está compuesta por una secuencia de líneas, donde en cada línea, está mínimo del vector leído en cada caso.

Ejemplo de entrada	Ejemplo de salida
2 4 10 -2.5 1.1 121 2 2.3 -2.3	-2.500000 -2.300000

1. Utilizar el tipo *float* de C
2. El tamaño máximo del vector es 2000 números reales

## Laboratorio 1d

Nombre del archivo fuente: lab01d.c

Tiempo límite: 1

Escribir un programa en C que lea un vector de números enteros y escriba la suma de los elementos del vector.

### Input

La entrada se debe tomar desde la entrada estandar. La entrada está compuesta por una secuencia de líneas, donde la primera línea contiene la cantidad de enteros a leer y las siguientes líneas contienen un entero. Como primera línea se encuentra un número que indica para cuantos vectores se debe realizar la suma de los elementos.

### Output

La salida debe enviarse a la salida estandar.

La salida está compuesta por una secuencia de líneas, donde en cada línea, está la suma de los elementos del vector leído en cada caso.

Ejemplo de entrada	Ejemplo de salida
2 4 10 -25 11 121 2 2 -2	117 0

1. Utilizar el tipo *int* de C
2. El tamaño máximo del vector es 2000 números enteros
3. Utilizar un procedimiento para leer el vector.
4. Utilizar una función para calcular la suma.

## Laboratorio 1e

Nombre del archivo fuente: lab01e.c

Tiempo limite: 1

Escribir un programa en C que lea dos números reales y escriba el resultado de usar los numeros como operandos de una operacion aritmética (suma, resta, multiplicación o división)

### Input

La entrada se debe tomar desde la entrada estandar. La entrada está compuesta por una secuencia de trios de líneas, donde la primera linea del trio tiene un caracter que indica la operacion (+, -, \*, /) las otras lineas del trio contienen los operandos. Como primera línea se encuentra un número que indica cuantas operaciones deben realizarse.

### Output

La salida debe enviarse a la salida estandar.

La salida está compuesta por una secuencia de líneas, donde en cada línea, está el resultado de aplicar la operación a los operandos leídos en cada caso

Ejemplo de entrada	Ejemplo de salida
5	7.500000
+	-119.900002
10	-5.290000
-2.5	1.500000
-	0.750000
1.1	
121	
*	
2.3	
-2.3	
/	
1.5	
1	
/	
1.5	
2	

Utilizar el tipo *float* de C