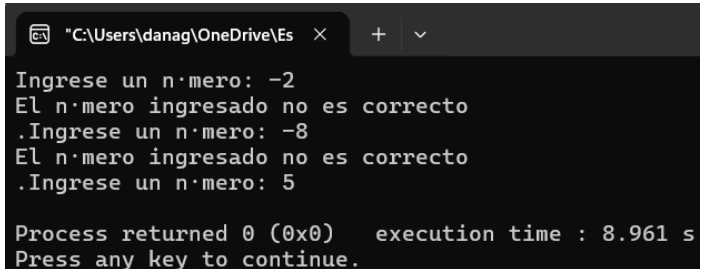


TRABAJO PRACTICO N2 - C+

EJERCICIO 1

A) Validar que un número sea mayor que cero. Seguir pidiendo si no cumple.

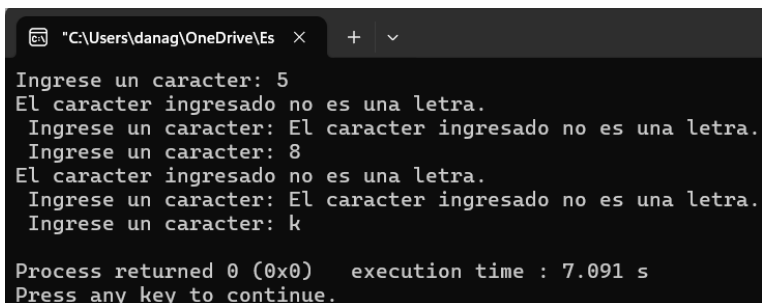
```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int LeerN()
5  {
6      int N;
7
8      do
9      {
10         printf("Ingrese un número: ");
11         scanf("%d", &N);
12         if(N <= 0)
13         {
14             printf("El número ingresado no es correcto\n.");
15         }
16     }
17     while(!(N>0)); //Mientras que no sea mayor a cero, seguir repitiendo
18
19     return(N);
20 }
21
22 int main()
23 {
24     int valor;
25
26     valor=LeerN();
27
28     return 0;
29 }
30
```



```
"C:\Users\danag\OneDrive\Escritorio"
Ingrese un número: -2
El número ingresado no es correcto
Ingrese un número: -8
El número ingresado no es correcto
Ingrese un número: 5
Process returned 0 (0x0)   execution time : 8.961 s
Press any key to continue.
```

B) Validar que un carácter sea una letra antes de retornarlo. Seguir pidiendo si no cumple.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <ctype.h>
4
5  char LeerL()
6  {
7      char L;
8
9      do
10     {
11         printf("Ingrese un caracter: ");
12         scanf("%c", &L);
13         if (!isalpha(L))
14         {
15             printf("El caracter ingresado no es una letra.\n ");
16         }
17     }
18     while(!(isalpha(L)));
19
20     return (L);
21 }
22
23 int main()
24 {
25     int valor;
26
27     valor = LeerL();
28
29     return 0;
30 }
31
```



```
"C:\Users\danag\OneDrive\Escritorio"
Ingrese un caracter: 5
El caracter ingresado no es una letra.
Ingrese un caracter: El caracter ingresado no es una letra.
Ingrese un caracter: 8
El caracter ingresado no es una letra.
Ingrese un caracter: El caracter ingresado no es una letra.
Ingrese un caracter: k
Process returned 0 (0x0)   execution time : 7.091 s
Press any key to continue.
```

ARREGLOS

EJERCICIO 2: Obtener la componente más grande de un vector de números enteros.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main()
6  {
7      int ar[10], i, MayorElem;
8
9      srand(665); //Cargo el arreglo
10     for(i=0; i<10; i++)
11         ar[i]=rand()%100; //Con dos dígitos
12
13     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
14
15     for(i=0; i<10; i++)
16         printf("%d ", ar[i]); //Muestro el arreglo
17
18     MayorElem=ar[0];
19
20     for(i=0; i<10; i++) //Recorro el arreglo
21     {
22         if(ar[i]>MayorElem)
23         {
24             MayorElem=ar[i];
25         }
26     }
27
28     printf("\n El valor mas grande es %d.", MayorElem);
29     return 0;
30 }
```

```
"C:\Users\danag\OneDrive\Es" x + v
Se acaba de cargar un arreglo con los siguientes numeros: 10 85 28 56 35 32 61 59 78 79
El valor mas grande es 85.
Process returned 0 (0x0)   execution time : 0.070 s
Press any key to continue.
```

EJERCICIO 3: Determinar cuántas veces aparece un valor entero N en las componentes de un vector de enteros V.

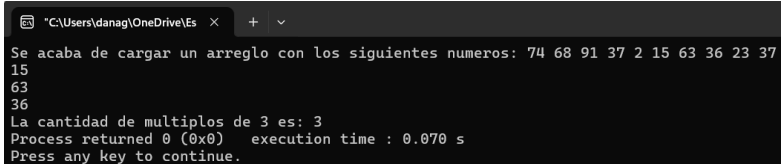
```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main()
6  {
7      int V[10], i, N, VecesElem;
8
9      srand(669); //Cargo el arreglo
10     for (i=0; i<10; i++)
11         V[i] = rand()%100;
12
13     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
14     for(i=0; i<10; i++)
15         printf("%d ", V[i]); //Muestro el arreglo
16
17     printf("\nIngrese un numero entero: ");
18     scanf("%d", &N); //Pido un N entero
19
20     VecesElem = 0; //Inicializo VecesElem en 0 porque es un contador, no quiero recorrer el vector como en el 2
21     for(i=0; i<10; i++)
22     {
23         if(N == V[i])
24         {
25             VecesElem = VecesElem + 1;
26         }
27     }
28
29     printf("La cantidad de veces que aparece %i en el arreglo es %d", N, VecesElem);
30
31     return 0;
32 }
```

```
"C:\Users\danag\OneDrive\Es" x + v
Se acaba de cargar un arreglo con los siguientes numeros: 23 11 49 6 25 39 11 50 88 55
Ingrese un numero entero: 55
La cantidad de veces que aparece 55 en el arreglo es 1
Process returned 0 (0x0)   execution time : 2.851 s
Press any key to continue.
```

EJERCICIO 5:

A) Contar y mostrar cuántos son múltiplos de 3 en un vector V de números aleatorios.

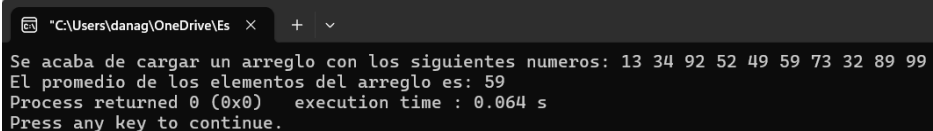
```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int V[10], i, cantmult;
7
8      srand(6656);
9      for (i=0; i<10; i++)
10         V[i] = rand()%100;
11
12     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
13     for(i=0; i<10; i++)
14         printf("%d ", V[i]);
15
16     cantmult = 0;
17
18     for(i=0; i<10; i++)
19     {
20         if(V[i]%3 == 0)
21         {
22             cantmult = cantmult + 1;
23             printf("\n%d", V[i]);
24         }
25     }
26     printf("\nLa cantidad de multiplos de 3 es: %d", cantmult);
27     return 0;
28 }
29
```



```
Se acaba de cargar un arreglo con los siguientes numeros: 74 68 91 37 2 15 63 36 23 37
La cantidad de multiplos de 3 es: 3
Process returned 0 (0x0)   execution time : 0.070 s
Press any key to continue.
```

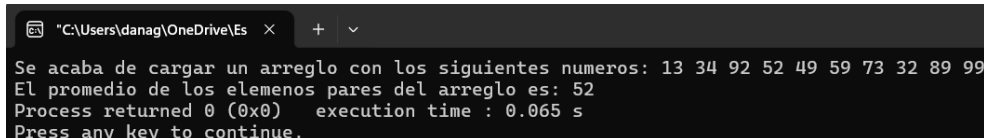
B) Calcular el promedio de los elementos de un vector V de números aleatorios.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4  #define tam 10
5
6  int main()
7  {
8      int V[tam], i, suma, prom;
9
10     srand(666);
11     for (i=0; i<tam; i++)
12         V[i] = rand()%100;
13
14     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
15     for(i=0; i<tam; i++)
16         printf("%d ", V[i]);
17
18     suma = 0;
19
20     for(i=0; i<tam; i++)
21     {
22         suma = suma + V[i];
23     }
24
25     prom = suma/tam;
26     printf("\nEl promedio de los elementos del arreglo es: %d", prom);
27
28     return 0;
29 }
30
```



```
Se acaba de cargar un arreglo con los siguientes numeros: 13 34 92 52 49 59 73 32 89 99
El promedio de los elementos del arreglo es: 59
Process returned 0 (0x0)   execution time : 0.064 s
Press any key to continue.
```

C) Calcular el promedio de los elementos pares de un vector de V números aleatorios.



```
Se acaba de cargar un arreglo con los siguientes numeros: 13 34 92 52 49 59 73 32 89 99
El promedio de los elementos pares del arreglo es: 52
Process returned 0 (0x0)   execution time : 0.065 s
Press any key to continue.
```

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main()
6  {
7      int V[10], i, cantpares, sumapares, prompares;
8
9      srand(666); //Cargo el arreglo con 2 DIGITOS (100)
10     for (i=0; i<10; i++)
11         V[i] = rand()%100;
12
13     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
14     for(i=0; i<10; i++)
15         printf("%d ", V[i]); //Muestro el arreglo
16
17     cantpares = 0; sumapares = 0;
18
19     for(i=0; i<10; i++)
20     {
21         if(V[i]%2 == 0)
22         {
23             cantpares = cantpares + 1;
24             sumapares = sumapares + V[i];
25         }
26     }
27
28     prompares = sumapares/cantpares;
29     printf("\nEl promedio de los elementos pares del arreglo es: %d", prompares);
30
31     return 0;
32 }

```

D) Determinar y mostrar si hay al menos un número múltiplo de 5 y no de 10 en un vector V de números aleatorios.

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <stdbool.h>
4
5  int main()
6  {
7      int V[10], i;
8      bool cumple;
9
10     srand(6766); //Cargo el arreglo con 2 DIGITOS (100)
11     for (i=0; i<10; i++)
12         V[i] = rand()%100;
13
14     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
15
16     for(i=0; i<10; i++)
17         printf("%d ", V[i]); //Muestro el arreglo
18
19     i = 0; cumple = false;
20
21     while(i<10) //El while recorre el arreglo (al menos 1 componente cumple)
22         //El for tambien recorre el arreglo (pero muestra todos los componentes que cumplen la condicion)
23     {
24         if((V[i]%5 == 0)&&(V[i]%10 != 0))
25         {
26             printf("\n%d ", V[i]); //Esto termina cuando: 1) Recorro todo el arreglo 2) Se cumple la condicion del while
27         }
28         i = i + 1; //Para moverme de i en i (en for se hace automatico, en while no, hay que agregarlo)
29     }
30
31     return 0;
32 }

```

```

"C:\Users\danag\OneDrive\Es x + v
Se acaba de cargar un arreglo con los siguientes numeros: 33 45 70 93 25 3 53 15 58 78
45
25
15
Process returned 0 (0x0) execution time : 0.064 s
Press any key to continue.

```

EJERCICIO 6: Calcular el producto escalar entre dos vectores de números aleatorios de máximo 2 dígitos.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main()
6  {
7      int V1[10], V2[10], i, suma;          //Los vectores tienen la misma cantidad de componentes
8
9      srand(666);                          //Cargo el arreglo 1
10     for (i=0; i<10; i++)
11         V1[i] = rand()%100;
12
13     printf("Se acaba de cargar un arreglo con los siguientes numeros: ");
14     for (i=0; i<10; i++)
15         printf("%d ", V1[i]);              //Muestro el arreglo 1
16
17     srand(667);                          //Cargo el arreglo 2
18     for (i=0; i<10; i++)
19         V2[i] = rand()%100;
20
21     printf("\nSe acaba de cargar un arreglo con los siguientes numeros: ");
22     for (i=0; i<10; i++)
23         printf("%d ", V2[i]);              //Muestro el arreglo 2
24
25     suma = 0;
26
27     for (i=0; i<10; i++)
28     {
29         suma = suma + V1[i]*V2[i];         //El producto escalar es igual a la sumatoria de las multiplicaciones
30                                             //de cada componente de ambos vectores  $u \cdot v = u_1 \cdot v_1 + u_2 \cdot v_2 + \dots + u_n \cdot v_n$ 
31     }
32     printf("\nEl producto escalar de ambos vectores es: %d", suma);
33     //El printf va afuera del for porque sino me lo muestra en cada paso del recorrido
34
35     return 0;
36 }
```

```
"C:\Users\danag\OneDrive\Es  x  +  v
Se acaba de cargar un arreglo con los siguientes numeros: 13 34 92 52 49 59 73 32 89 99
Se acaba de cargar un arreglo con los siguientes numeros: 16 14 88 15 96 86 86 5 33 51
El producto escalar de ambos vectores es: 33762
Process returned 0 (0x0)   execution time : 0.078 s
Press any key to continue.
```