

# Guía 1

## Programación I

Rodrigo Estay

Nombre en Timus: RodrigoEstay

Todos los ejercicios a continuación fueron escritos sin ayuda.

1.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  /* Revisamos en cada numero del 2 al 10 si es divisor, si lo es, lo imprimimos. */
5
6  int main (){
7      int num;
8      printf("Ingrese un numero:\n");
9      scanf("%d", &num);
10     printf("Divisores entre 2 y 10:\n");
11     if(num%2==0)
12         printf("2 ");
13     if(num%3==0)
14         printf("3 ");
15     if(num%4==0)
16         printf("4 ");
17     if(num%5==0)
18         printf("5 ");
19     if(num%6==0)
20         printf("6 ");
21     if(num%7==0)
22         printf("7 ");
23     if(num%8==0)
24         printf("8 ");
25     if(num%9==0)
26         printf("9 ");
27     if(num%10==0)
28         printf("10 ");
29     printf("\n");
30     return 0;
31 }
```

2.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 /* Ya que son solo 2 numeros, existen solo 2 casos de cual sea mayor. */
5
6 int main (){
7     int num1, num2;
8     printf("Ingrese dos numeros enteros:\n");
9     scanf("%d%d", &num1, &num2);
10    if(num2>num1)
11        printf("El doble del mayor:\n%d\n", num2*2);
12    else
13        printf("El doble del mayor:\n%d\n", num1*2);
14    return 0;
15 }
```

3.

```
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 /* Es un poco confusa la sentencia "el primero restado del segundo" pero espero
5 que sea el segundo menos el primero, en el programa solo aplicamos la formula,
6 t en el caso de que son iguales, comparamos si el modulo de division es mayor
7 a 0, en tan caso se cumple invariante. */
8
9 int main (){
10     int num1, num2, num3;
11     printf("Ingrese 3 numeros:\n");
12     scanf("%d%d%d", &num1, &num2, &num3);
13     if((num2-num1<(num3*num3)/num1) || ((num2-num1==(num3*num3)/num1) && (0<(num3*num3)%num1)))
14         printf("Cumple invariante\n");
15     return 0;
16 }
```

4.

```
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 /* Ocupamos la variable "temp" para almacenar el numero que va a ser el primero
5 en cambiarse, asi al cambiarlo por el otro numero no se pierde el dato guardado
6 por el primero. consiguiendo un swap exitoso. */
7
8 int main (){
9     int num1, num2, temp;
10    printf("Ingrese 2 numeros:\n");
11    scanf("%d%d", &num1, &num2);
12    if((num1<0 && num2>=0) || (num2<0 && num1>=0)){
13        printf("Intercambiando numeros...\n");
14        temp=num1;
15        num1=num2;
16        num2=temp;
17    }
18    else
19        printf("No se cumple criterios para intercambio.\n");
20    printf("num1=%d\nnum2=%d\n", num1, num2);
21    return 0;
22 }
```

5.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  /* No hay mucha ciencia, solo se revisan los numeros ingresados a ver si cumple
5  uno de todos los casos posibles. */
6
7  int main (){
8      int num1, num2, num3;
9      printf("Introduzca 3 numeros:\n");
10     scanf("%d%d%d", &num1, &num2, &num3);
11     if (num1<0 && num2<0 && num3<0)
12         printf("todos\n");
13     else if (num1>=0 && num2>=0 && num3>=0)
14         printf("ninguno\n");
15     else if (num1>=0 && num2<0 && num3<0)
16         printf("al menos dos\n");
17     else if (num1<0 && num2>=0 && num3<0)
18         printf("al menos dos\n");
19     else if (num1<0 && num2<0 && num3>=0)
20         printf("al menos dos\n");
21     else if (num1>=0 && num2>=0 && num3<0)
22         printf("al menos uno\n");
23     else if (num1<0 && num2>=0 && num3>=0)
24         printf("al menos uno\n");
25     else if (num1>=0 && num2<0 && num3>=0)
26         printf("al menos uno\n");
27     else
28         printf("ERROR\n");
29     return 0;
30 }
```

6.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  /* Si es que los 3 numeros no cumplen que sean todos pares o impares, si o si se
5  cumplira el caso de que 2 son impares o 2 pares. */
6
7  int main (){
8      int num1, num2, num3;
9      printf("Introduzca 3 numeros:\n");
10     scanf("%d%d%d", &num1, &num2, &num3);
11     if((num1%2==0 && num2%2==0 && num3%2==0) || (num1%2==1 && num2%2==1 && num3%2==1))
12         printf("uniforme.\n");
13     else
14         printf("variadito.\n");
15     return 0;
16 }
```

7.

```
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 /* Primero separamos los numeros en pares que comparamos y asi definimos un par
5 mayor y uno menor, luego comparamos estos pares para saber que numero es el
6 primero, el ultimo, y los de en medio, finalmente comparamos los del medio
7 para obtener el orden de menor a mayor. */
8
9 int main (){
10     int num1, num2, num3, num4;
11     printf("Introduzca 4 numeros:\n");
12     scanf("%d%d%d%d", &num1, &num2, &num3, &num4);
13     int mayor1, mayor2, menor1, menor2, enmedio1, primero, ultimo, enmedio2;
14     printf("Los numeros de menor a mayor:\n");
15     if(num1>num2)
16         mayor1=num1, menor1=num2;
17     else
18         menor1=num1, mayor1=num2;
19     if(num3>num4)
20         mayor2=num3, menor2=num4;
21     else
22         menor2=num3, mayor2=num4;
23     if(mayor1>mayor2)
24         primero=mayor1, enmedio1=mayor2;
25     else
26         enmedio1=mayor1, primero=mayor2;
27     if(menor1>menor2)
28         enmedio2=menor1, ultimo=menor2;
29     else
30         ultimo=menor1, enmedio2=menor2;
31     if(enmedio1<enmedio2)
32         printf("%d %d %d %d\n", ultimo, enmedio1, enmedio2, primero);
33     else
34         printf("%d %d %d %d\n", ultimo, enmedio2, enmedio1, primero);
35     return 0;
36 }
```

8.

```
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 /* Ocupamos un switch, asi saltamos entre casos e imprimimos el numero de
5 de veces indicado. */
6
7 int main (){
8     int num;
9     printf("Ingrese un numero entre 1 y 10:\n");
10    scanf("%d", &num);
11    if(num<=0 || num>10){
12        printf("ERROR NUMERO INGRESADO INVALIDO, PROGRAMA TERMINADO.\n");
13        return 0;
14    }
15    switch(num){
16        case 10:
17            printf("paraguacutilimicuario\n");
18        case 9:
19            printf("paraguacutilimicuario\n");
20        case 8:
21            printf("paraguacutilimicuario\n");
22        case 7:
23            printf("paraguacutilimicuario\n");
24        case 6:
25            printf("paraguacutilimicuario\n");
26        case 5:
27            printf("paraguacutilimicuario\n");
28        case 4:
29            printf("paraguacutilimicuario\n");
30        case 3:
31            printf("paraguacutilimicuario\n");
32        case 2:
33            printf("paraguacutilimicuario\n");
34        case 1:
35            printf("paraguacutilimicuario\n");
36    }
37    return 0;
38 }
```

9.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  /* Se saca el area (A*B) luego se multiplica por la cantidad de paneles (*N)
5  luego se multiplica por 2 ya que se debe aplicar el proceso por los dos lados
6  de los paneles. */
7
8  int main (){
9      int N, A, B;
10     scanf("%d%d%d", &N, &A, &B);
11     printf("%d\n", A*B*N*2);
12     return 0;
13 }
```

10.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  /* Tenemos que a las latas que les falto disparar a cada uno es igual al numero
5  de latas que el otro le disparo menos una lata que es a la que los dos le
6  dispararon a la vez. */
7
8  int main (){
9      int h, l;
10     scanf("%d%d", &h, &l);
11     printf("%d %d\n", l-1, h-1);
12     return 0;
13 }
```

11.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  /* Solo usamos un if para cada caso dado por la tabla. */
5
6  int main (){
7      int n;
8      scanf("%d", &n);
9      if(1<=n && n<=4)
10         printf("few\n");
11     if(5<=n && n<=9)
12         printf("several\n");
13     if(10<=n && n<=19)
14         printf("pack\n");
15     if(20<=n && n<=49)
16         printf("lots\n");
17     if(50<=n && n<=99)
18         printf("horde\n");
19     if(100<=n && n<=249)
20         printf("throng\n");
21     if(250<=n && n<=499)
22         printf("swarm\n");
23     if(500<=n && n<=999)
24         printf("zounds\n");
25     if(1000<=n)
26         printf("legion\n");
27     return 0;
28 }
```

12.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  /* Tenemos que existen 4 casos, cuando hay 0 bistecs, mas espacio para bistecs en la sartén que la cantidad de
5  bistecs, cuando los bistecs que sobran son menores o igual que la mitad de los bistecs que caben en la sartén
6  y cuando es el caso contrario al anterior. */
7
8  int main(){
9      int n, k;
10     scanf("%d%d", &n, &k);
11     if(n==0)
12         printf("0\n");
13     else if(k>n)
14         printf("2\n");
15     else if(n%k==0)
16         printf("%d\n", (n*2)/k);
17     else if(n%k<=k/2)
18         printf("%d\n", (n/k)*2+1);
19     else
20         printf("%d\n", (n/k)*2+2);
21     return 0;
22 }
23
24 /* Cabe destacar que en los ultimos dos casos lo que ocurre es que si sobran bistecs en la penultima "cocinada"
25 y los que sobran son menor o igual a la mitad de lo que cabe en la sartén, se puede optimizar el tiempo de
26 cocción de estos ultimos dos grupos, ya que reemplazando bistecs a medio cocer con los crudos, y luego
27 sacando los ya cocidos por los que sacamos antes, ahorraremos un minuto, quitando un minuto a la formula
28 del penultimo caso.
29
30 y a la formula se le suma 2 minutos ya que como en n/k se trunca al entero, tendremos que perdemos un minuto
31 gracias a la formula, y luego al multiplicar por dos terminamos perdiendo 2 minutos, los cuales se deben sumar
32 al final. */
```

13.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  /* Como el ladron parte del codigo 0000 y empieza a subir, y ademas que el dueño
5  de la bicicleta cambia el seguro todos los días entre 2 seguros y ademas parte
6  con el seguro 1, podemos darnos cuenta de que el ladron siempre intentara con
7  numeros pares para el seguro 1 y con numeros impares para el seguro 2, entonces
8  si estos no cumplen esas condiciones, el ladron nunca robara la bicicleta. */
9
10 int main (){
11     int lock1, lock2;
12     scanf("%d%d", &lock1, &lock2);
13     if(lock1%2==0 || lock2%2==1)
14         printf("yes\n");
15     else
16         printf("no\n");
17     return 0;
18 }
```

14.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  /* Es tan simple como ver el signo al restarle a 240 minutos los minutos que
5  le tomara hacer los problemas restantes, si da negativo no alcanza, si da
6  positivo si alcanza. */
7
8  int main (){
9     int f, queda;
10     scanf("%d", &f);
11     queda=4*60-((12-f)*45);
12     if(queda>=0)
13         printf("YES\n");
14     else
15         printf("NO\n");
16     return 0;
17 }
```