

1. Leer números enteros de teclado, hasta que el usuario ingrese el 0. Finalmente, mostrar la sumatoria de todos los números ingresados.

The screenshot shows the PSeInt IDE with a program named 'positivos y negativos suma.psc'. The program code is as follows:

```
1 Algoritmo Nenteros
2   definir n, sumar Como entero;
3   n=1;
4   sumar=0;
5   Mientras n#0 Hacer
6     Escribir "ingrese un numero.";
7     leer n;
8     si n#0 Entonces
9       sumar=sumar+n;
10    SiNo
11      Escribir "finalizo el programa";
12    FinSi
13  FinMientras
14  Escribir "la suma es ",sumar;
15 FinAlgoritmo
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
ingrese un numero.
> 12
ingrese un numero.
> 33
ingrese un numero.
> 23
ingrese un numero.
> 15
ingrese un numero.
> 0
finalizo el programa
la suma es 83
*** Ejecución Finalizada. ***
```

2. Leer números enteros de teclado, hasta que el usuario ingrese el 0. Finalmente, mostrar la sumatoria de todos los números positivos ingresados.

The screenshot shows the PSeInt IDE with a program named 'sin_titulo.psc'. The program code is as follows:

```
1 Algoritmo sin_titulo
2   definir n , nn, np Como Entero;
3   n = 0;
4   nn = 0;
5   np = 0;
6   Escribir "ingrese un numero";
7   leer n;
8   Mientras n # 0 Hacer
9     si n > 0 Entonces
10      np = np+n;
11      Escribir np;
12    SiNo
13      si n = 0 Entonces
14        Escribir "Salir";
15      FinSi
16    FinSi
17    Escribir "ingrese nuevo numero positivo:";
18    leer n;
19  FinMientras
20  FinAlgoritmo
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
ingrese nuevo numero positivo:
> 23
23
ingrese nuevo numero positivo:
> 234
257
ingrese nuevo numero positivo:
> -2
ingrese nuevo numero positivo:
> 123
380
ingrese nuevo numero positivo:
> 0
*** Ejecución Finalizada. ***
```

3. Leer números enteros positivos de teclado, hasta que el usuario ingrese el 0. Informar cuál fue el mayor número ingresado.

The screenshot shows the PSeInt IDE with a Pascal program named 'comparar.psc'. The program uses a 'Mientras' loop to read positive integers until the user enters 0, then prints the maximum value. The execution window shows the program running with inputs 34, 234, and 1892, resulting in the output 'el numero mayor es 1892'.

```
1 Algoritmo comparar
2   definir n,almacen Como Entero;
3   n:=1;
4   almacen:=0;
5   Mientras n # 0 Hacer
6     Escribir "ingrese un numero positivo";
7     leer n;
8     Si n>0 Entonces
9       si n>almacen Entonces
10        almacen:=n;
11      FinSi
12    SiNo
13      si n=0 Entonces
14        | escribir "ingrese un numero positivo";
15      FinSi
16    Fin Si
17  Fin Mientras
18  Escribir "el numero mayor es ",almacen;
19 FinAlgoritmo
```

Execution Output:

```
*** Ejecución Iniciada. ***
ingrese un numero positivo
> 34
ingrese un numero positivo
> 234
ingrese un numero positivo
> 1892
ingrese un numero positivo
> 0
el numero mayor es 1892
*** Ejecución Finalizada. ***
```

4. Leer un número entero positivo desde teclado e imprimir la suma de los dígitos que lo componen.

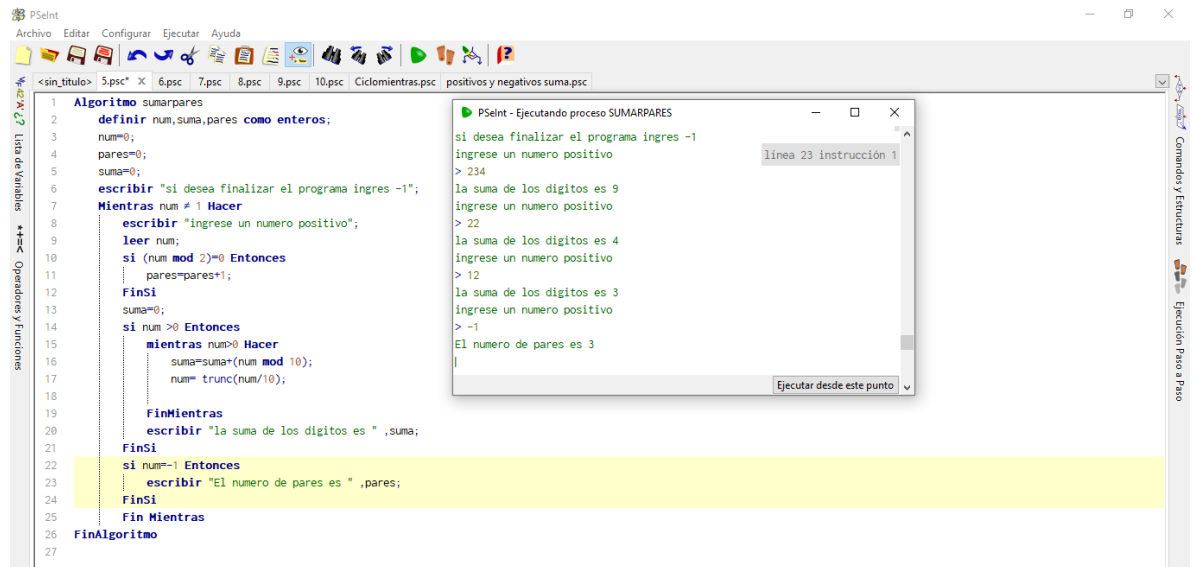
The screenshot shows the PSeInt IDE with a Pascal program named 'sumardigitos.psc'. The program reads a positive integer and calculates the sum of its digits using a loop. The execution window shows the program running with input 1234, resulting in the output 'La suma de los digitos es 10'.

```
1 Algoritmo SumarDigitos
2   definir n,suma Como Entero;
3   Imprimir "ingresa un numero entero";
4   leer n;
5   suma:=0;
6   Mientras n>0 Hacer
7     suma=suma+(n mod 10);
8     n:= trunc(n/10);
9   Fin Mientras
10  Imprimir "La suma de los digitos es ",suma;
11 FinAlgoritmo
```

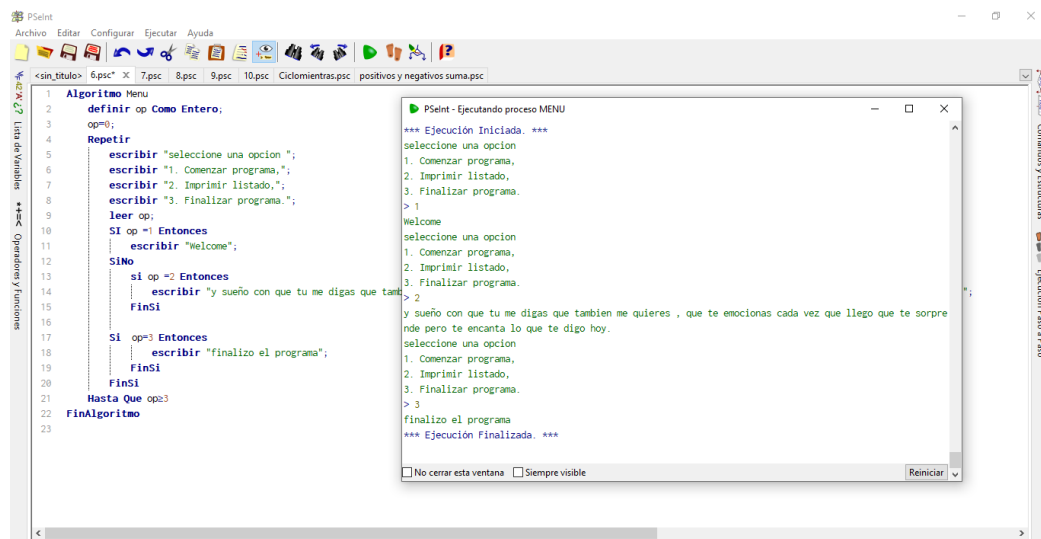
Execution Output:

```
*** Ejecución Iniciada. ***
ingresa un numero entero
> 1234
La suma de los digitos es 10
*** Ejecución Finalizada. ***
```

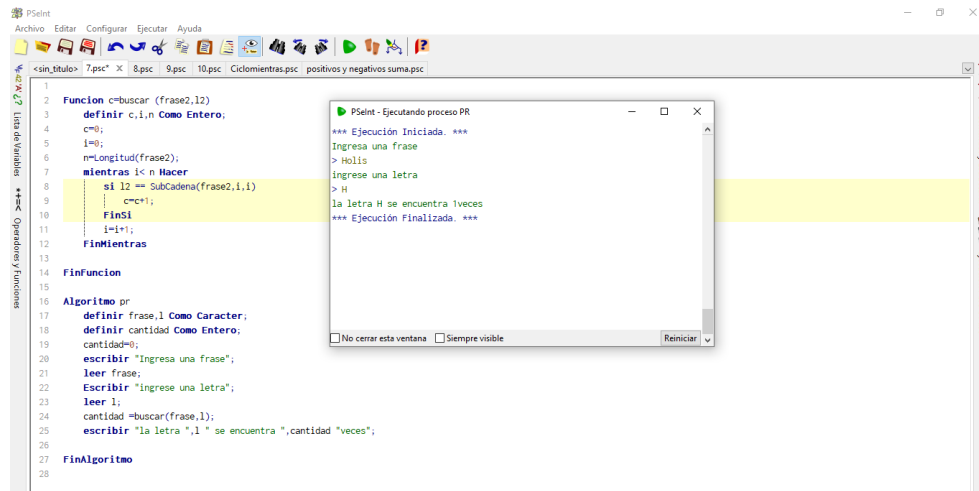
5. Solicitar al usuario que ingrese números enteros positivos y, por cada uno, imprimir la suma de los dígitos que lo componen. La condición de corte es que se ingrese el número - 1. Al finalizar, mostrar cuántos de los números ingresados por el usuario fueron números pares



6. Mostrar un menú con tres opciones: 1- comenzar programa, 2- imprimir listado, 3- finalizar programa. A continuación, el usuario debe poder seleccionar una opción (1, 2 ó 3). Si elige una opción incorrecta, informarle del error. El menú se debe volver a mostrar luego de ejecutada cada opción, permitiendo volver a elegir. Si elige las opciones 1 ó 2 se imprimirá un texto. Si elige la opción 3, se interrumpirá la impresión del menú y el programa finalizará.

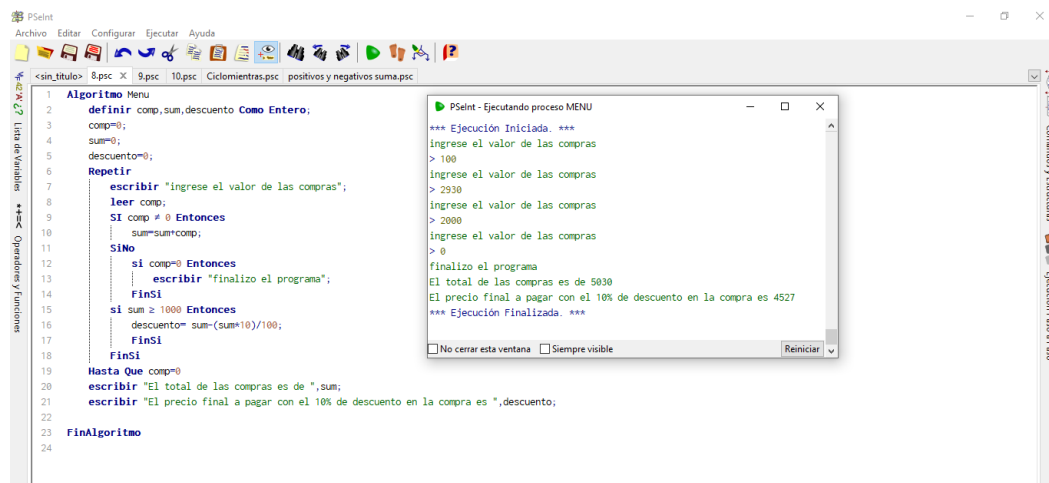


7. Solicitar al usuario el ingreso de una frase y de una letra (que puede o no estar en la frase). Recorrer la frase, carácter a carácter, comparando con la letra buscada. Si el carácter no coincide, indicar que no hay coincidencia en esa posición (imprimiendo la posición) y continuar. Si se encuentra una coincidencia, indicar en qué posición se encontró y finalizar la ejecución.



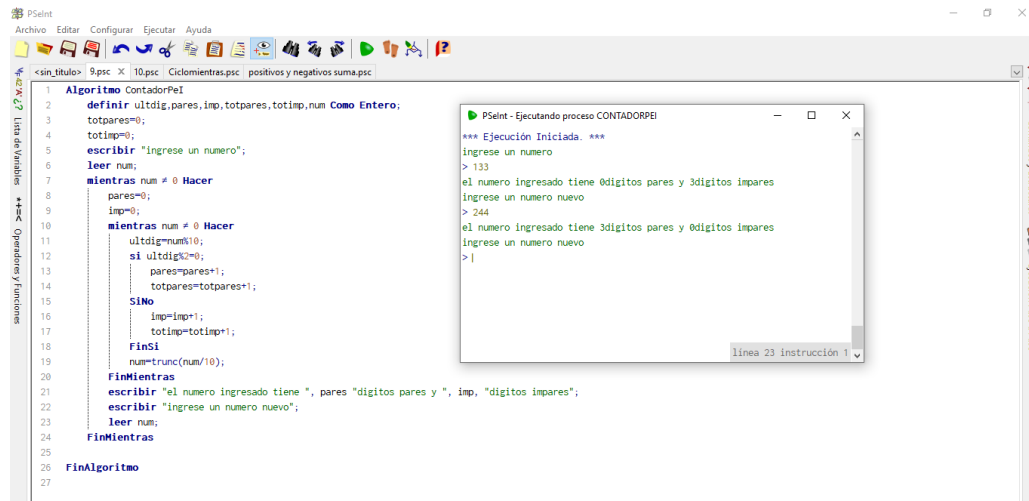
```
1  Funcion c=buscar (frase2,l2)
2  definir c,l,n Como Entero;
3  c=0;
4  l=0;
5  n=Longitud(frase2);
6  mientras l<n Hacer
7  si l2 == SubCadena(frase2,l,l)
8  c=c+1;
9  FinSi
10 l=l+1;
11 FinMientras
12 FinFuncion
13
14 Algoritmo pr
15 definir frase,l Como Caracter;
16 definir cantidad Como Entero;
17 cantidad=0;
18 escribir "Ingresar una frase";
19 leer frase;
20 escribir "Ingresar una letra";
21 leer l;
22 cantidad=buscar(frase,l);
23 escribir "la letra ",l," se encuentra ",cantidad," veces";
24
25 FinAlgoritmo
```

8. Crear un programa que permita al usuario ingresar los montos de las compras de un cliente (se desconoce la cantidad de datos que cargará, la cual puede cambiar en cada ejecución), cortando el ingreso de datos cuando el usuario ingrese el monto 0. Si ingresa un monto negativo, no se debe procesar y se debe pedir que ingrese un nuevo monto. Al finalizar, informar el total a pagar teniendo que cuenta que, si las ventas superan el total de \$1000, se le debe aplicar un 10% de descuento.



```
1  Algoritmo Menu
2  definir comp,sum,descuento Como Entero;
3  comp=0;
4  sum=0;
5  descuento=0;
6  Repetir
7  escribir "ingrese el valor de las compras";
8  leer comp;
9  SI comp <= 0 Entonces
10 sum=sum+comp;
11 SINO
12 si comp<0 Entonces
13 escribir "finalizo el programa";
14 FinSi
15 si sum >= 1000 Entonces
16 descuento= sum*(sum*10)/100;
17 FinSi
18 FinSi
19 Hasta Que comp=0
20 escribir "El total de las compras es de ",sum;
21 escribir "El precio final a pagar con el 10% de descuento en la compra es ",descuento;
22
23 FinAlgoritmo
```

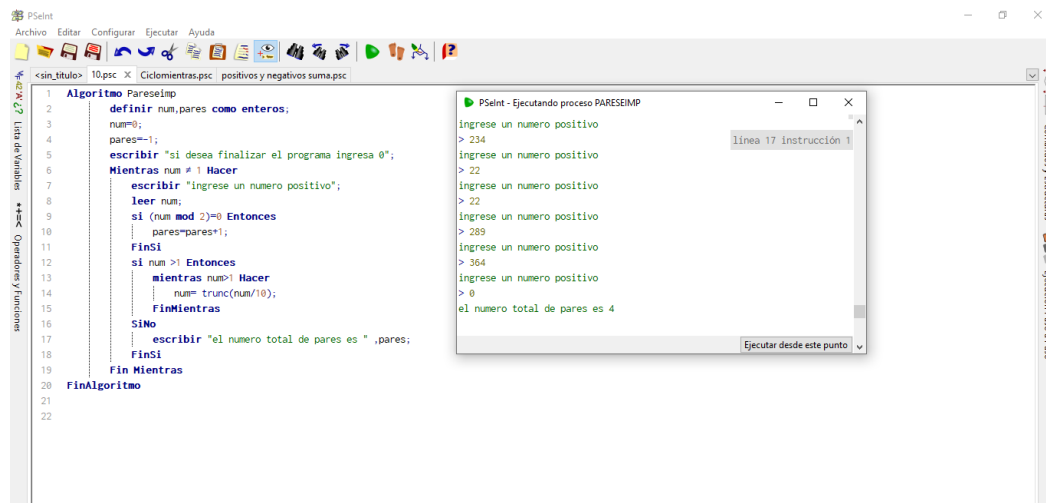
9. Crear un programa que solicite el ingreso de números enteros positivos, hasta que el usuario ingrese el 0. Por cada número, informar cuántos dígitos pares y cuántos impares tiene. Al finalizar, informar la cantidad de dígitos pares y de dígitos impares leídos en total.



The screenshot shows the PSeInt IDE with a Pascal program named 'ContadorPeI'. The program uses a 'Mientras' loop to process numbers until the user enters 0. It counts the number of even and odd digits for each input. A console window titled 'PSeInt - Ejecutando proceso CONTADORPEI' shows the execution flow, including inputs like 133 and 244, and the resulting counts of even and odd digits.

```
1 Algoritmo ContadorPeI
2   definir ultdig, pares, imp, tot pares, tot imp, num Como Entero;
3   tot pares:=0;
4   tot imp:=0;
5   escribir "ingrese un numero";
6   leer num;
7   mientras num ≠ 0 Hacer
8     pares:=0;
9     imp:=0;
10    mientras num ≠ 0 Hacer
11      ultdig:=num%10;
12      si ultdig%2=0;
13        pares:=pares+1;
14        tot pares:=tot pares+1;
15      SiNo
16        imp:=imp+1;
17        tot imp:=tot imp+1;
18      FinSi
19      num:=trunc(num/10);
20    FinMientras
21    escribir "el numero ingresado tiene ", pares "digitos pares y ", imp, "digitos impares";
22    escribir "ingrese un numero nuevo";
23    leer num;
24  FinMientras
25
26 FinAlgoritmo
```

10. Escribir un programa que solicite el ingreso de una cantidad indeterminada de números mayores que 1, finalizando cuando se reciba un cero. Imprimir la cantidad de números primos ingresados



The screenshot shows the PSeInt IDE with a Pascal program named 'Pareselimp'. The program uses a 'Mientras' loop to process numbers until the user enters 0. It checks if each number is prime and counts the total number of primes. A console window titled 'PSeInt - Ejecutando proceso PARESEIMP' shows the execution flow, including inputs like 234, 22, 289, 364, and 0, and the final output: 'el numero total de pares es 4'.

```
1 Algoritmo Pareselimp
2   definir num, pares como enteros;
3   num:=0;
4   pares:=1;
5   escribir "si desea finalizar el programa ingresa 0";
6   Mientras num ≠ 1 Hacer
7     escribir "ingrese un numero positivo";
8     leer num;
9     si (num mod 2)=0 Entonces
10      pares:=pares+1;
11    FinSi
12    si num > 1 Entonces
13      mientras num > 1 Hacer
14        num:=trunc(num/10);
15      FinMientras
16    SiNo
17      escribir "el numero total de pares es " , pares;
18    FinSi
19  FinMientras
20
21 FinAlgoritmo
```