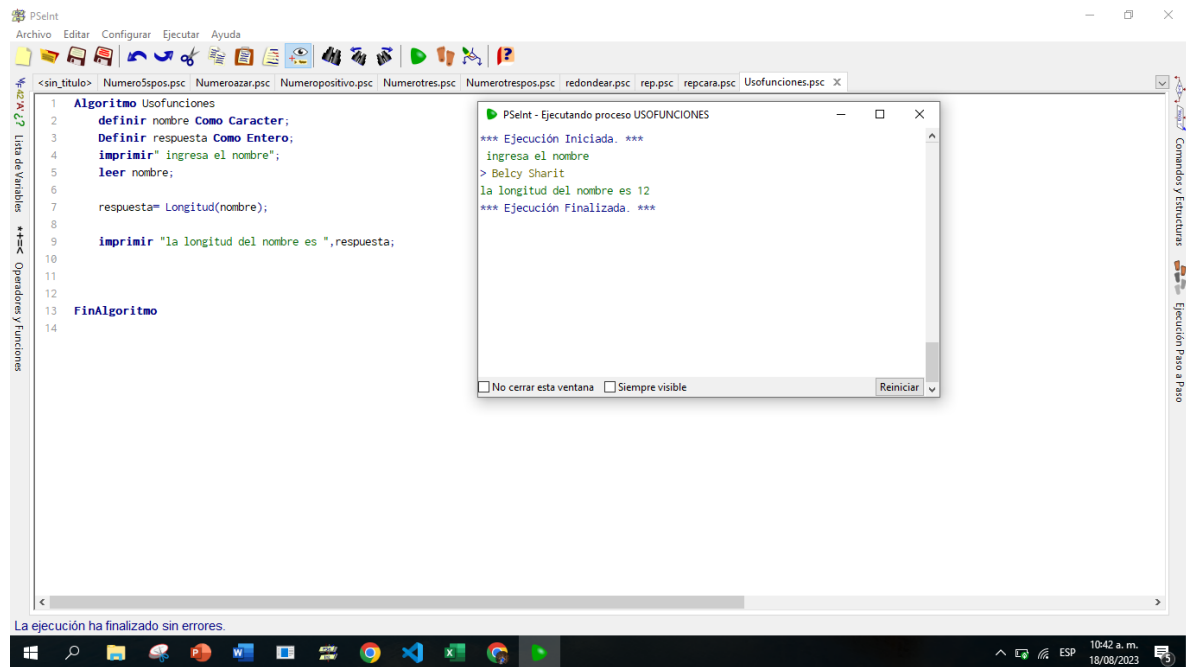
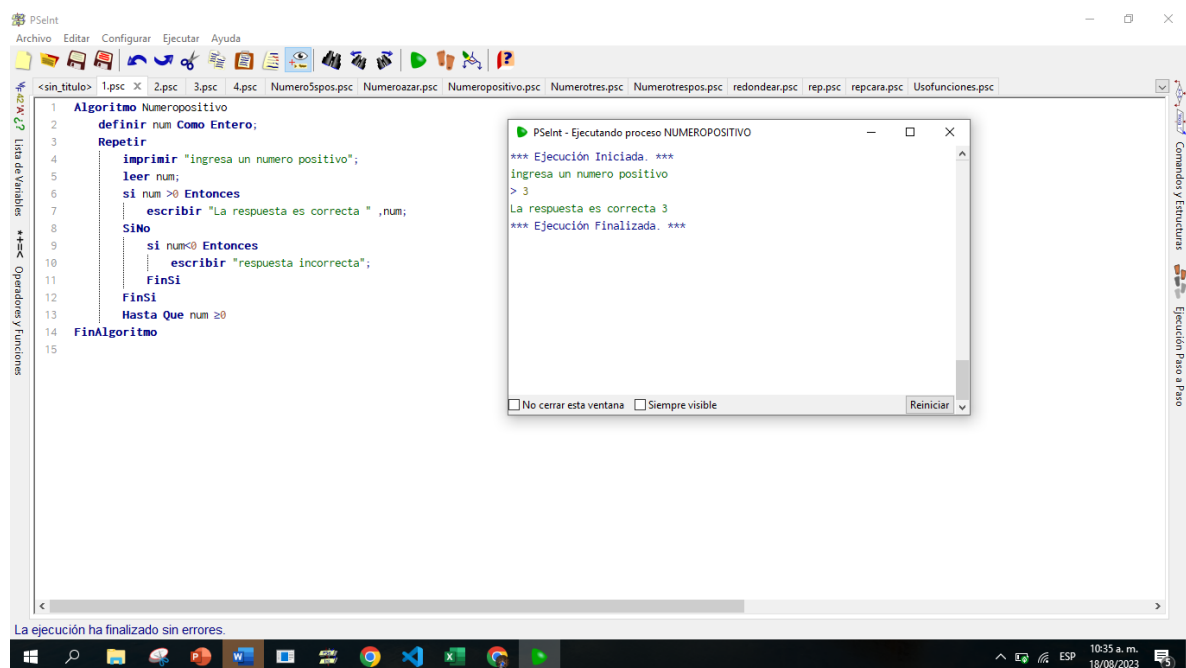


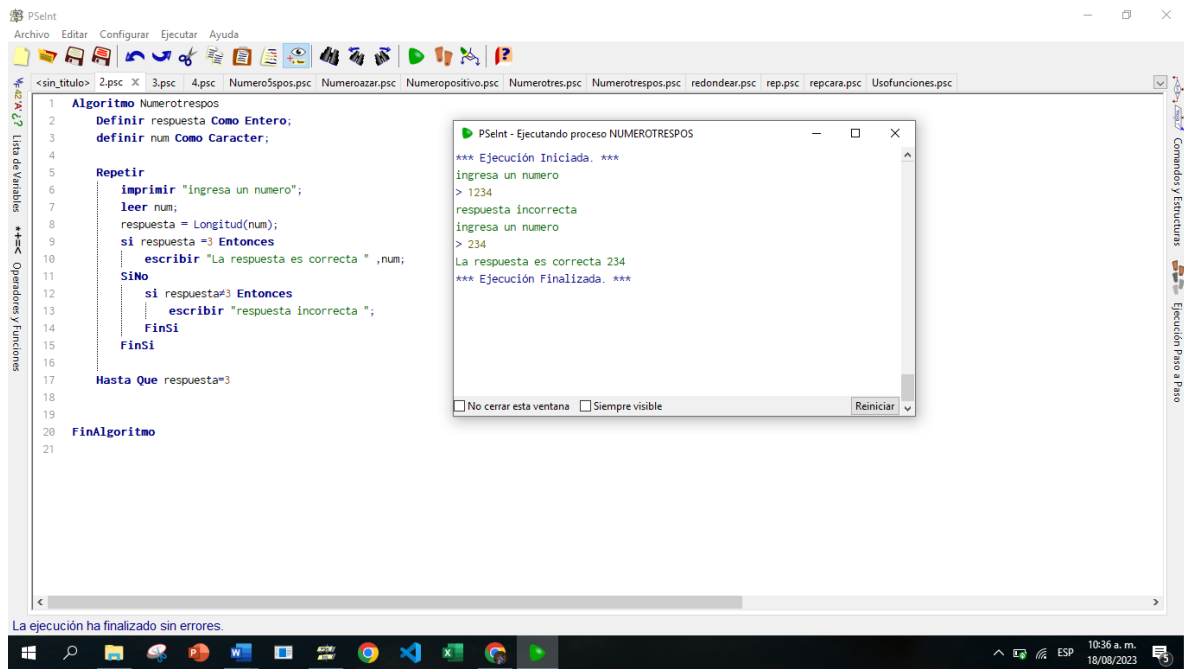
## Usos funciones



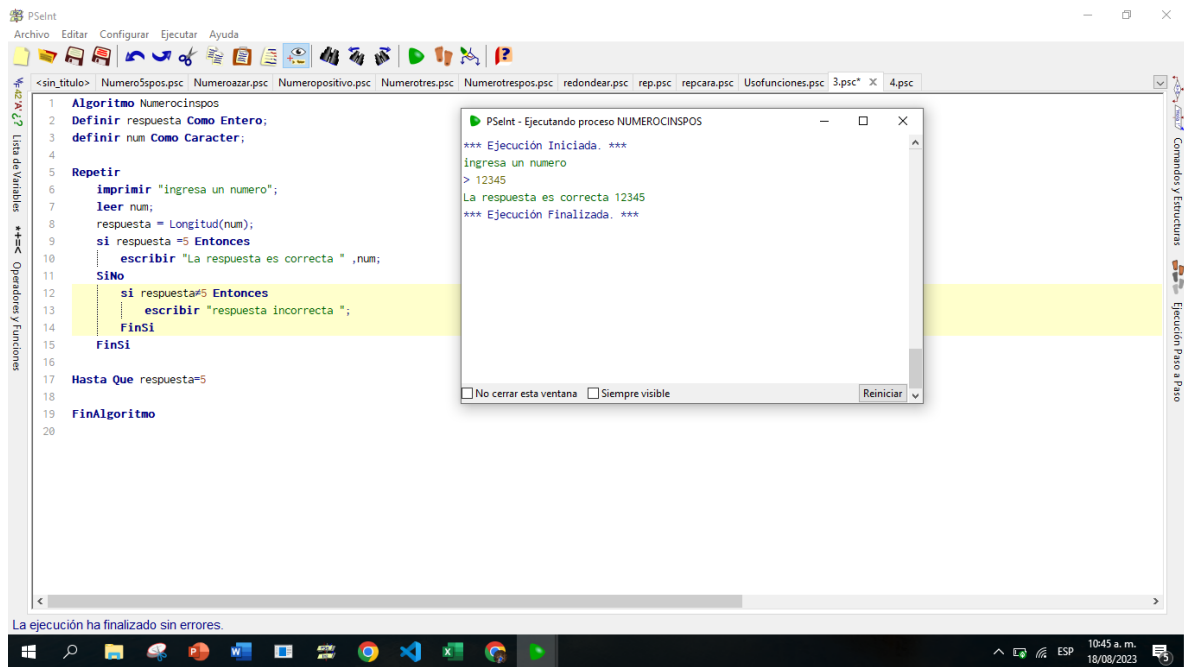
1.



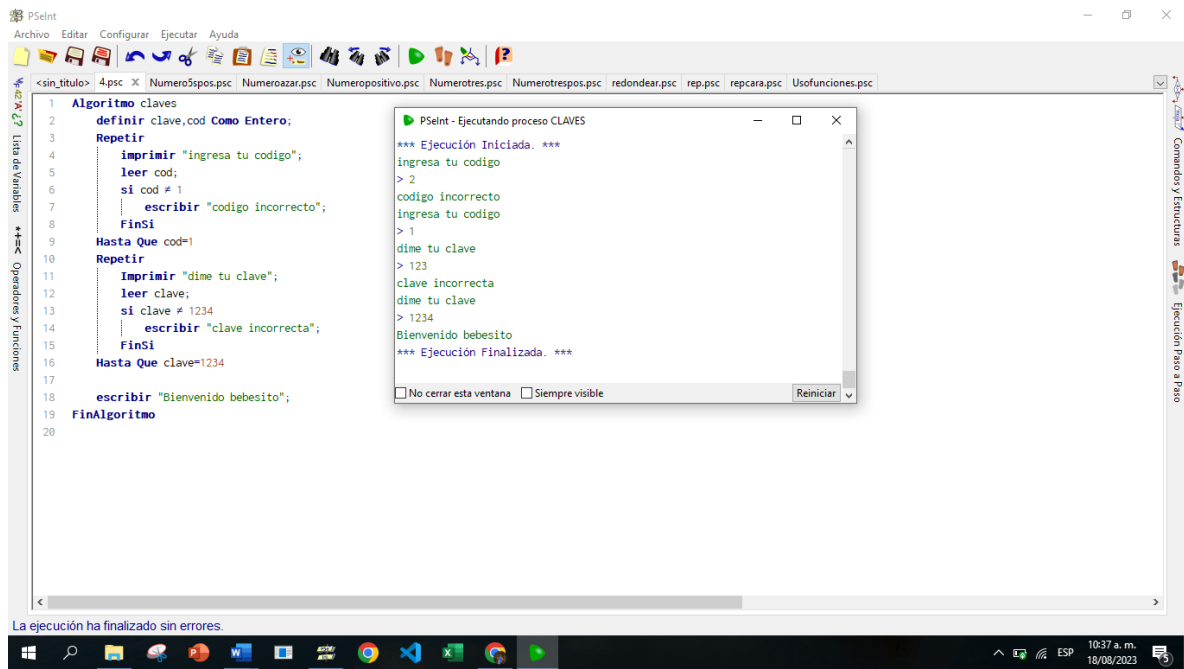
2.



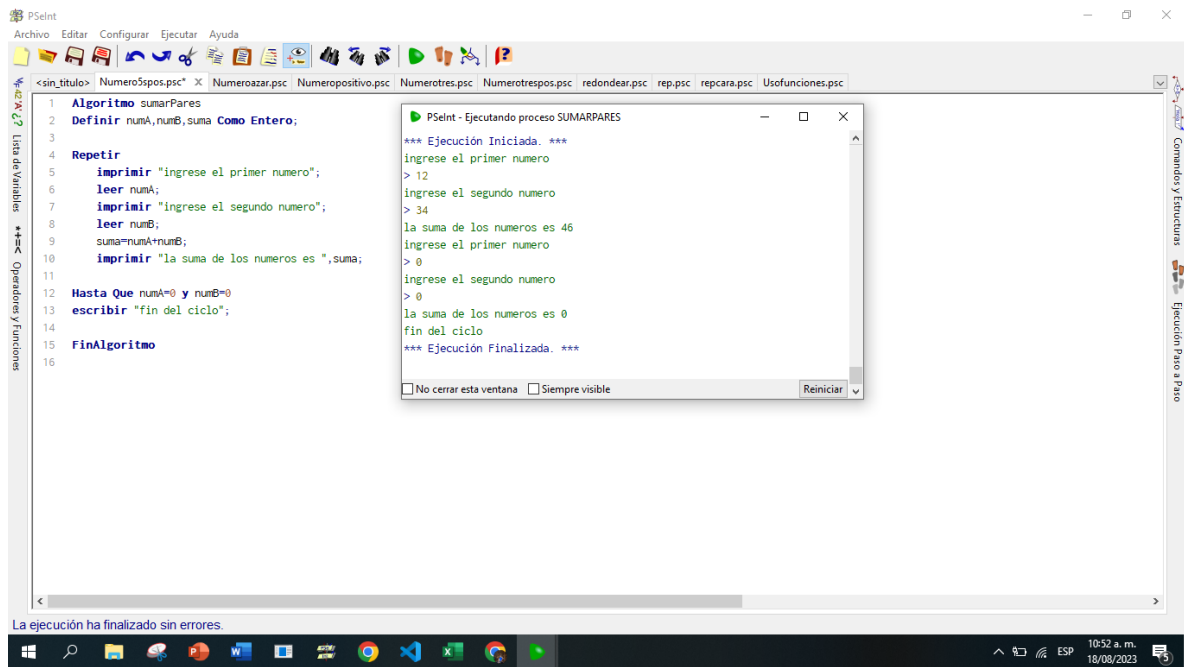
3.



4.



5.



6.

La ejecución ha finalizado sin errores.

7.

La ejecución ha finalizado sin errores.

8.

El pseudocódigo está siendo ejecutado.

```

1  Algoritmo MCM
2  definir numA,numB,x,c,a,b Como Entero;
3  Escribir "ingrese dos numeros";
4  leer numA,numB;
5  x=0;
6  si numA>numB Entonces
7  | x=numB;
8  SiNo
9  | si x=numA entonces
10 | | numA=numB;
11 | FinSi
12 FinSi
13 numB=1;
14 Repetir
15 | a=numA*numB;
16 | numB=numB+1;
17 | b=0;
18 | c=1;
19 | mientras b<a Hacer
20 | | b=x*c;
21 | | c=c+1;
22 | | si a=b Entonces
23 | | | Escribir "el minimo comun multiplo es ",a;
24 | | | x=0;
25 | | FinSi
26 | FinMientras
27 Hasta Que numB=0
28
29 FinAlgoritmo
  
```

PSeInt - Ejecutando proceso MCM

```

*** Ejecución Iniciada. ***
ingrese dos numeros
> 24
> 8
el minimo comun multiplo es 24
  
```

9.

La ejecución ha finalizado sin errores.

```

1  Algoritmo Nperfect
2  definir num,x,perfect Como Entero;
3  escribir "ingrese un numero";
4  leer num;
5  x=2;
6  perfect=0;
7  si num<0 Entonces
8  | Repetir
9  | | si num mod x=0 Entonces
10 | | | perfect=perfect+(num/x);
11 | | FinSi
12 | | x=x+1;
13 | Hasta Que x>num
14 | si perfect=num Entonces
15 | | escribir "el numero ",num " es perfecto";
16 | SiNo
17 | | escribir "el numero ",num " no es perfecto";
18 | FinSi
19 SiNo
20 | escribir "el numero debe ser mayor a 0";
21 FinSi
22
23
24 FinAlgoritmo
25
  
```

PSeInt - Ejecutando proceso NPERFECT

```

*** Ejecución Iniciada. ***
ingrese un numero
> 28
el numero 28 es perfecto
*** Ejecución Finalizada. ***
  
```

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar

10.

