



universidad
cenfotec_
La U de la informática

Principios de Programación.

Ejemplos resueltos de listas.

Ejemplos resueltos de Funciones

A continuación, se presentan algunos de los ejemplos de listas.

Primer ejemplo de listas

Ejemplo 1:

Problema:

Se quiere saber cuál es el promedio de las notas de un grupo de estudiantes de Principios de Programación.

1. Análisis.

2. Diseño o modelado.

3. Implementación.

Definición:

Se supone que las notas del grupo de estudiantes de Principios de Programación se almacenan en una lista de números reales, así se pueden sumar todos los elementos de la lista y luego se divide la suma entre el número de estudiantes para obtener el promedio, pero hay que tener en cuenta que no se sabe con anterioridad cuántos estudiantes hay en el grupo. El número de estudiantes es dado por el usuario.

Ejemplo 1:

Variables de entrada y salida:

1. Análisis.
2. Diseño o modelado.
3. Implementación.

Variables del programa			
Descripción	Notación		Ejemplo
	Nombre	Tipo	
Entrada			
Variable para guardar el número de estudiantes del grupo.	cant_estudiantes	entero	3
Variable para guardar todas las notas del grupo de estudiantes	lista_notas []	lista	[78, 90.5, 89.4]
Intermedias			
Variable que guarda la suma de las notas.	suma_total	float	257.9
Salida			
Variable que almacena el promedio de las notas del grupo.	promedio	float	85.96

Ejemplo 1:

Planteamiento

En el lista **notas** se van a almacenar las notas de todos los estudiantes del grupo de Principios de Programación, pero no se sabe con anterioridad cuántos estudiantes hay en el grupo. Es por esto que se deben seguir los siguientes pasos:

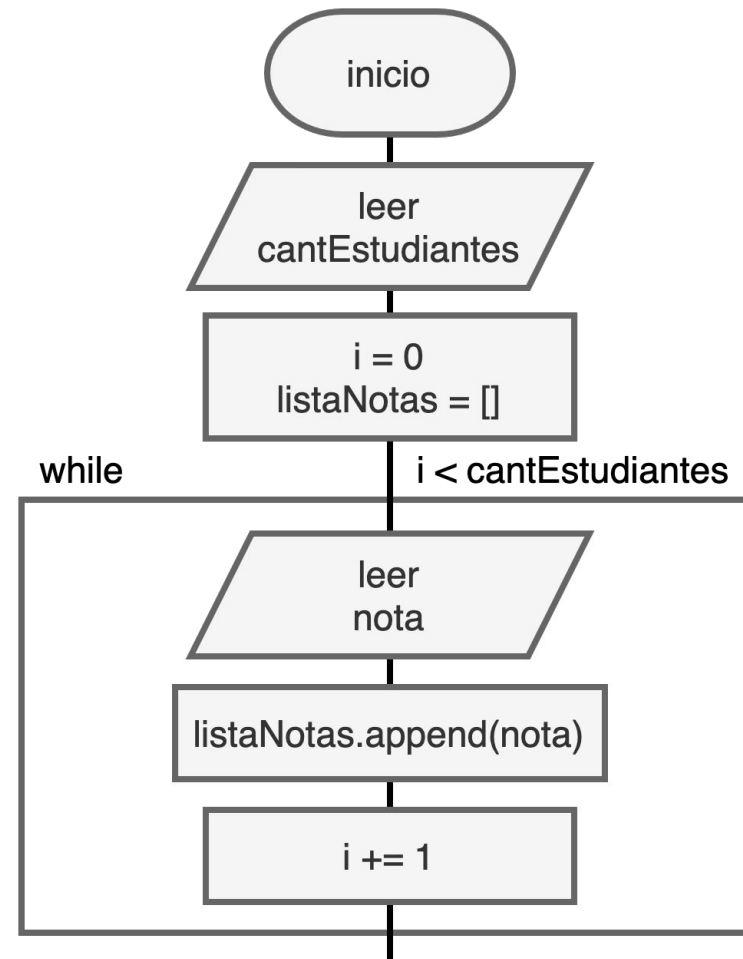
1. Análisis.
 2. Diseño o modelado.
 3. Implementación.
1. Declarar el lista de reales llamado **notas**.
 2. Preguntarle al usuario cuántos estudiantes tiene el grupo y reservar el espacio en memoria para ello.
 3. Leer el lista **notas**.
 4. Calcular el promedio de las notas del curso e imprimirlo.

Para leer el lista se usa una **estructura iterativa con contador**, y luego otra para calcular la suma de todas las notas, por último, se divide la suma de todas las notas entre el número de estudiantes para poder calcular el promedio.

Ejemplo 1:

Diagrama de flujo:
Segmento del programa
para leer el lista.

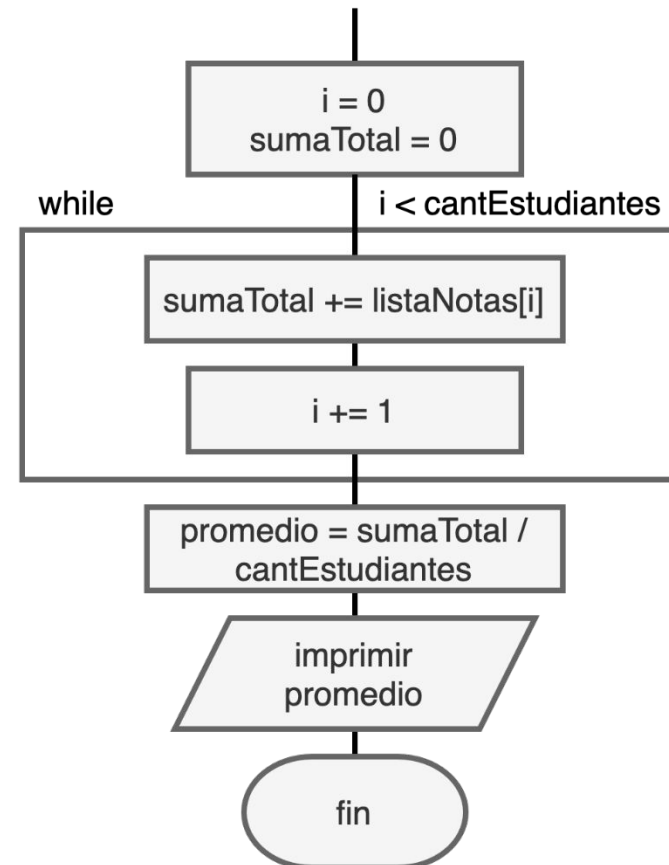
1. Análisis.
2. Diseño o modelado.
3. Implementación.



Ejemplo 1:

Diagrama de flujo:

Segmento del programa para calcular el promedio.



1. Análisis.
2. Diseño o modelado.
3. Implementación.

Ejemplo 1:

Implementación de la solución en código

1. Análisis.
2. Diseño o modelado.
3. Implementación.

```
cant_estudiantes = int(input('Ingrese la cantidad de estudiantes: '))
i = 0
lista_notas = []
while(i < cant_estudiantes):
    nota = float(input(f'Ingrese la nota {i + 1}: '))
    lista_notas.append(nota)
    i += 1

i = 0
suma_total = 0
while(i < cant_estudiantes):
    suma_total += lista_notas[i]
    i += 1
promedio = suma_total / cant_estudiantes
print(f'El promedio de notas es de : {promedio}')
```

Nota: En la segunda estructura iterativa del cálculo del promedio, en la evaluación del ciclo se puede reemplazar a **cant_estudiantes** por **len(lista_notas)**

Segundo ejemplo de listas

Ejemplo 2:

Problema:

Se quiere saber cuál es la mayor venta que se registró en un almacén entre todas las ventas registradas en un año, en el que se guarda el total de ventas de cada mes.

1. Análisis.

2. Diseño o modelado.

Definición:

3. Implementación.

Se supone que las ventas del almacén registradas en un año se almacenan en una lista de números reales de 12 posiciones, una para cada mes del año de ventas que se quiere registrar. Así se pueden revisar todos los elementos de la lista para poder hallar la mayor venta.

Ejemplo 2:

Variables de entrada y salida:

1. Análisis.
2. Diseño o modelado.
3. Implementación.

Variables del programa			
Descripción	Notación		Ejemplo
	Nombre	Tipo	
Entrada			
Variable para guardar todas las ventas de los 12 meses.	listaVentas []	lista	[78000,..., 89500]
Salida			
Variable que el valor asociado a la mayor venta realizada.	mayorVenta	float	96000

Ejemplo 2:

Planteamiento

En el lista **ventas** se van a almacenar las ventas del almacén que corresponden a cada mes. Para poder hallar la mayor venta se deben seguir los siguientes pasos:

1. Análisis.
2. Diseño o modelado.
 1. Declarar el lista de reales llamado **ventas** y reservar el espacio en memoria para 12 posiciones, una para cada mes.
 2. Leer el lista **ventas**.
 3. Calcular la mayor venta e imprimirla.
3. Implementación.

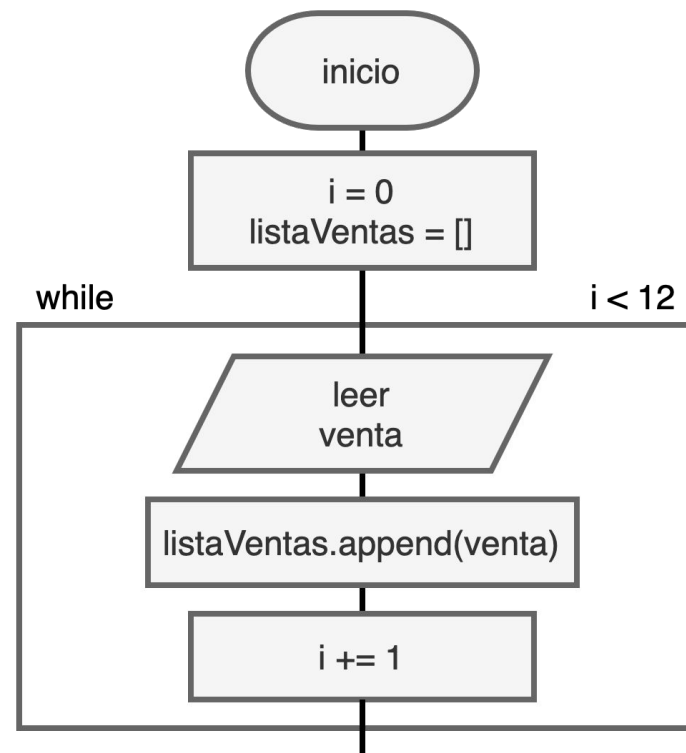
Para leer el lista se usa una **estructura iterativa con contador**, y otra para calcular la mayor venta. Para calcular la mayor venta se supone que al inicio la mayor venta fue en el primer mes, es decir, el primer elemento de la lista. Luego se recorre el lista para comparar con cada elemento de la lista, si se encuentra algún elemento que es mayor que la mayor venta, se cambia su valor por ese elemento. Por último, se imprime la mayor venta.

Ejemplo 2:

Diagrama de flujo:

Segmento del programa para leer el lista.

1. Análisis.
2. Diseño o modelado.
3. Implementación.

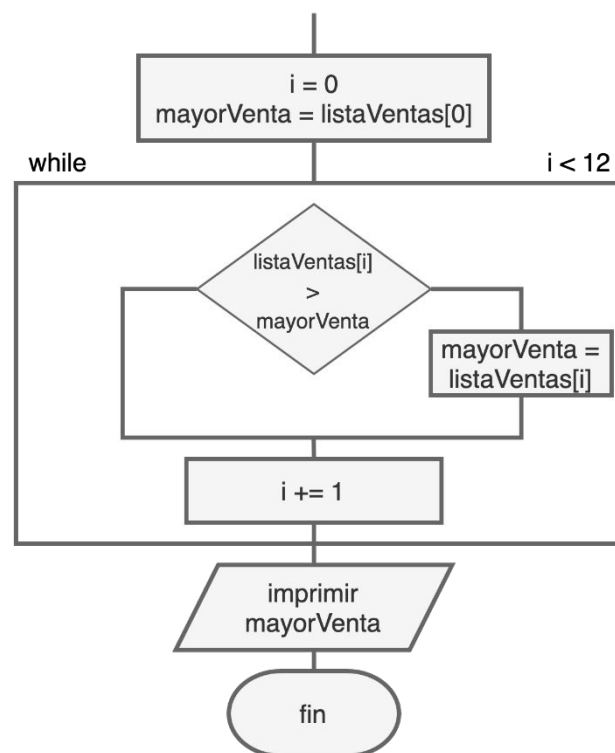


Ejemplo 2:

Diagrama de flujo:

Segmento del programa para hallar la mayor venta.

1. Análisis.
2. Diseño o modelado.
3. Implementación.



Ejemplo 2:

Implementación de la solución en código

1. Análisis.
2. Diseño o modelado.
3. Implementación.

```
i = 0
lista_ventas = []
while(i < 12):
    venta = float(input('Ingrese la venta del mes '))
    lista_ventas.append(venta)
    i += 1
i = 0
mayor_venta = lista_ventas[0]

while(i < 12):
    if(lista_ventas[i] > mayor_venta):
        mayor_venta = lista_ventas[i]
    i += 1
print(f'La mayor venta en el año fue de: {mayor_venta}')
```




universidad
cenfotec_
La U de la informática