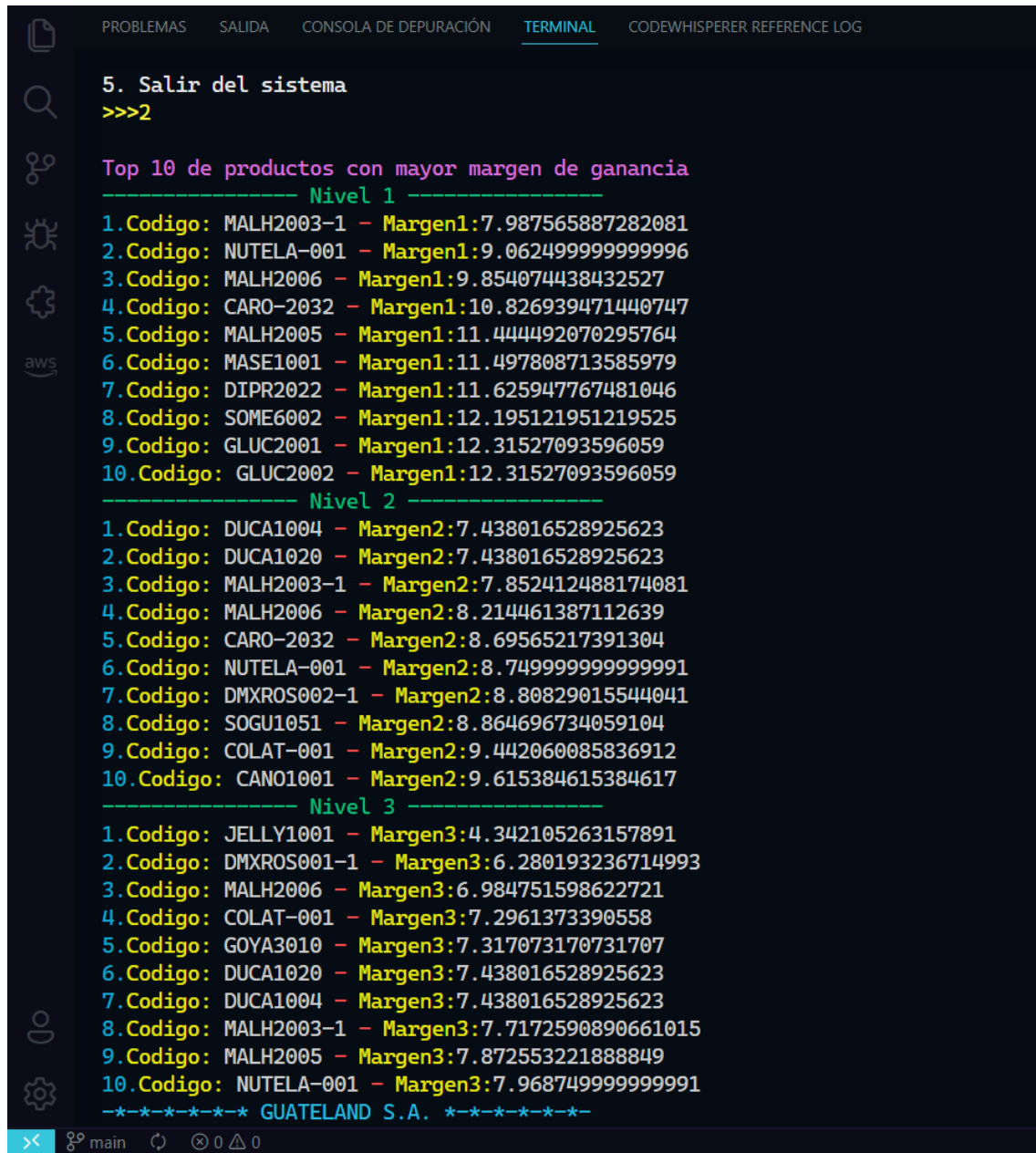


PRACTICA AUXILIAR

• Link: https://github.com/fabio-hernandez/IPC2_PRACTICA_201801005

• Pantallazos:



```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  CODEWHISPERER REFERENCE LOG

5. Salir del sistema
>>>2

Top 10 de productos con mayor margen de ganancia
----- Nivel 1 -----
1.Codigo: MALH2003-1 - Margen1:7.987565887282081
2.Codigo: NUTELA-001 - Margen1:9.062499999999996
3.Codigo: MALH2006 - Margen1:9.854074438432527
4.Codigo: CARO-2032 - Margen1:10.826939471440747
5.Codigo: MALH2005 - Margen1:11.444492070295764
6.Codigo: MASE1001 - Margen1:11.497808713585979
7.Codigo: DIPR2022 - Margen1:11.625947767481046
8.Codigo: SOME6002 - Margen1:12.195121951219525
9.Codigo: GLUC2001 - Margen1:12.31527093596059
10.Codigo: GLUC2002 - Margen1:12.31527093596059
----- Nivel 2 -----
1.Codigo: DUCA1004 - Margen2:7.438016528925623
2.Codigo: DUCA1020 - Margen2:7.438016528925623
3.Codigo: MALH2003-1 - Margen2:7.852412488174081
4.Codigo: MALH2006 - Margen2:8.214461387112639
5.Codigo: CARO-2032 - Margen2:8.69565217391304
6.Codigo: NUTELA-001 - Margen2:8.749999999999999
7.Codigo: DMXROS002-1 - Margen2:8.80829015544041
8.Codigo: SOGU1051 - Margen2:8.864696734059104
9.Codigo: COLAT-001 - Margen2:9.442060085836912
10.Codigo: CANO1001 - Margen2:9.615384615384617
----- Nivel 3 -----
1.Codigo: JELLY1001 - Margen3:4.342105263157891
2.Codigo: DMXROS001-1 - Margen3:6.280193236714993
3.Codigo: MALH2006 - Margen3:6.984751598622721
4.Codigo: COLAT-001 - Margen3:7.2961373390558
5.Codigo: GOYA3010 - Margen3:7.317073170731707
6.Codigo: DUCA1020 - Margen3:7.438016528925623
7.Codigo: DUCA1004 - Margen3:7.438016528925623
8.Codigo: MALH2003-1 - Margen3:7.7172590890661015
9.Codigo: MALH2005 - Margen3:7.872553221888849
10.Codigo: NUTELA-001 - Margen3:7.968749999999999
--*--*--*--* GUATELAND S.A. *--*--*--*--*

x main 0 0 0
```

-----* GUATELAND S.A. *-----

1. Cargar archivo para procesamiento
2. Obtener TOP 10 de productos con mayor margen de ganancias
3. Obtener TOP 10 de productos con mayor valor de inventario
4. Información del estudiante
5. Salir del sistema

>>>3

Top 10 productos con mayor valor de Inventario

- 1.Codigo: SOGU1028 - valorInventario:-272.25
- 2.Codigo: BOCA1005 - valorInventario:-120.89999999999999
- 3.Codigo: DUCA4004 - valorInventario:-42.6
- 4.Codigo: CARO-0064 - valorInventario:-32.96
- 5.Codigo: SOGU1010 - valorInventario:-24.06
- 6.Codigo: DELF2004 - valorInventario:-21.15
- 7.Codigo: INA2001 - valorInventario:-15.47
- 8.Codigo: LAK2010 - valorInventario:-10.05
- 9.Codigo: IDE003 - valorInventario:0.0
- 10.Codigo: DIAN3001 - valorInventario:0.0

-----* GUATELAND S.A. *-----

1. Cargar archivo para procesamiento
2. Obtener TOP 10 de productos con mayor margen de ganancias
3. Obtener TOP 10 de productos con mayor valor de inventario
4. Información del estudiante
5. Salir del sistema

>>>



main* 0 0

- Código Lista Enlazada y ordenamiento:

```
class Nodo:
    def __init__(self, dato=None, siguiente=None):
        self.dato = dato
        self.siguiente = siguiente

class listaSimple:

    def __init__(self):
        self.cabeza = None
        self.ultimo = None
        self.tamano = 0

    def insertar(self, dato):
        if self.cabeza is None:
            nuevo = Nodo(dato)
            self.cabeza = nuevo
            self.ultimo = nuevo
            self.tamano += 1
        else:
            nuevo = Nodo(dato)
            self.ultimo.siguiente = nuevo
            self.ultimo = self.ultimo.siguiente
            self.tamano += 1

    def ordenar(self, llave):
        aux = self.cabeza
        while aux:
            minimo = aux
            siguiente = aux.siguiente
            while siguiente:
                if float(getattr(siguiente.dato, llave)) < float(getattr(minimo.dato, llave)):
                    minimo = siguiente
                siguiente = siguiente.siguiente
            if aux != minimo:
                aux.dato, minimo.dato = minimo.dato, aux.dato
                aux = aux.siguiente
        self.imprimir10(llave)
```

- Código cálculo de márgenes:

```
def calcularMargenes(self, precio, costo):
    aux = precio - costo
    margen = (aux / costo) * 100
    return margen

def calcularValorInventario(self, cantidad, costo):
    valor = cantidad * costo
    return valor
```