

Desarrollo Laboratorio 2 - Estructuras Discretas

II Semestre 2023

Estudiante: Joy Nelaton

Cedula: 8-902-1282

Problema 1.

```
6  ## Problema 1
7
8  def fibloop(n):
9      lista =[0,1] ## Creacion de la lista
10     while len(lista)<n: ##ciclo while mientras la longtud de lista sea menor a n
11         lista.append(lista[-1]+ lista[-2]) ## se anexa a la lista los dos elementos anteriores
12         print(lista[:10]) ## se imprime la lista
13     return lista[:n] ## retorno de la lista
14
15
16  fibloop(10) ## llamado de la funcion
17
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

 Python Debug

AS/LAB_2_MAYO_4_2023/PARTES/1.py

```
[0, 1, 1]
[0, 1, 1, 2]
[0, 1, 1, 2, 3]
[0, 1, 1, 2, 3, 5]
[0, 1, 1, 2, 3, 5, 8]
[0, 1, 1, 2, 3, 5, 8, 13]
[0, 1, 1, 2, 3, 5, 8, 13, 21]
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

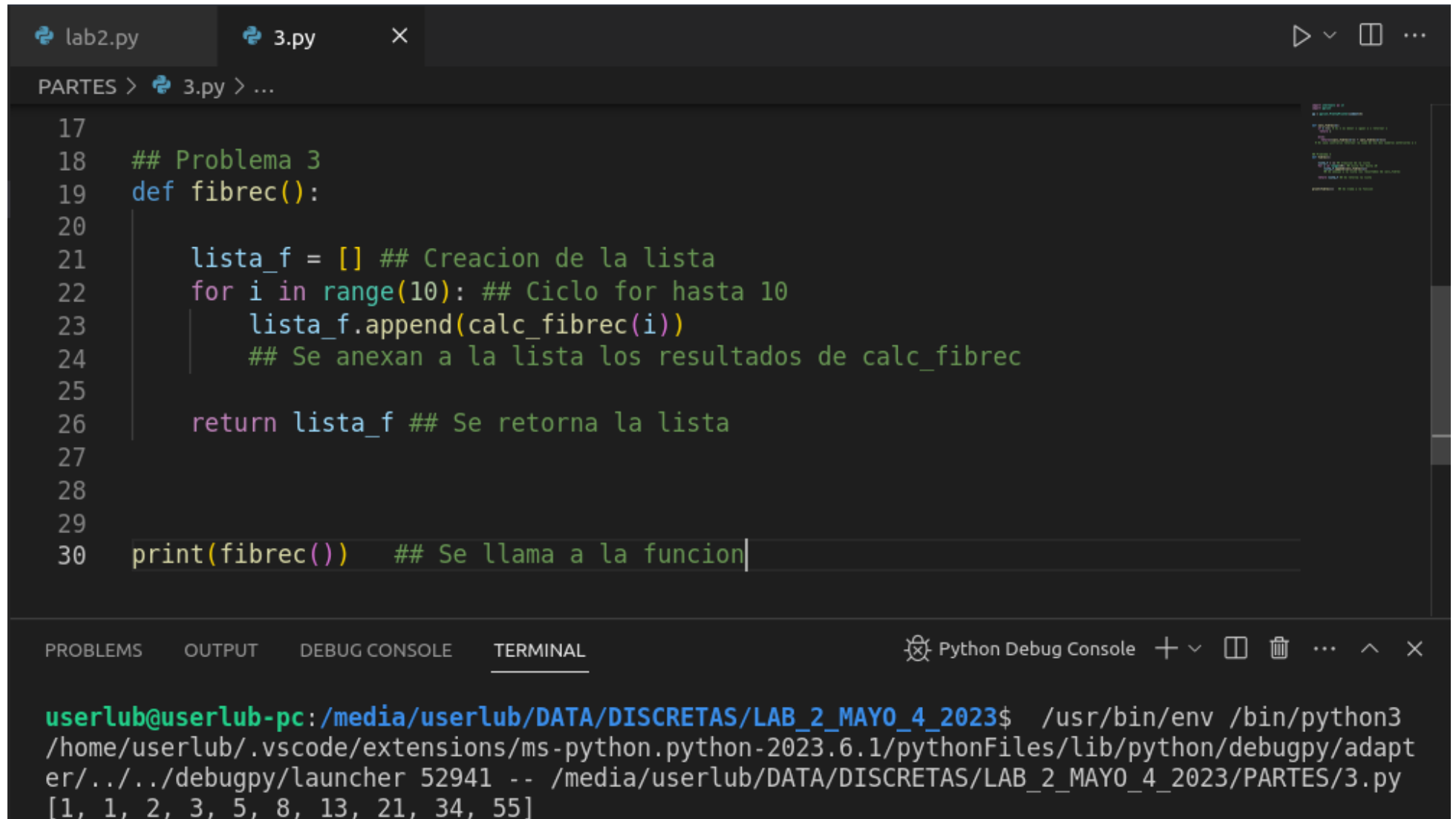
Problema 2.

```
2.py > calc_fibrec
1  import itertools as it
2  import pprint
3
4  pp = pprint.PrettyPrinter(indent=4)
5
6
7  ## Problema 2
8  def calc_fibrec(n):
9
10     if n <=1: # Si n es menor o igual a 1 retornar 1
11         return 1
12
13     else:
14         return((calc_fibrec(n-1) + calc_fibrec(n-2)))
15     # En caso contrario retornar la suma de los dos numeros anteriores a n
16
17
18  print(calc_fibrec(25)) # Llamado de la funcion para conocer el termino 25
19
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 44509 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/2.py
121393
```

Problema 3.



The image shows a VS Code editor window with two tabs: 'lab2.py' and '3.py'. The '3.py' tab is active, displaying a Python script. The script defines a function 'fibrec()' that creates a list 'lista_f' and appends the results of 'calc_fibrec(i)' for 'i' in the range 0 to 9. It then prints the result of 'fibrec()'. The bottom panel shows the 'TERMINAL' output, which includes the command to run the script and the resulting list of Fibonacci numbers: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55].

```
17
18  ## Problema 3
19  def fibrec():
20
21      lista_f = [] ## Creacion de la lista
22      for i in range(10): ## Ciclo for hasta 10
23          lista_f.append(calc_fibrec(i))
24          ## Se anexan a la lista los resultados de calc_fibrec
25
26      return lista_f ## Se retorna la lista
27
28
29
30  print(fibrec())  ## Se llama a la funcion
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 52941 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/PARTES/3.py
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
```

Problema 4.

```
4.py > euler_totient
7 def euler_totient(n):
8
9     r = n
10    p = 2
11
12    while p * p <= n: #Ciclos repetitivos para iterar entre numeros primos p
13        if n%p == 0:
14            while n % p == 0:
15                n //= p
16                r *= 1 - (1/p)
17            p = p + 1
18
19    if n > 1:
20        r *= 1 - (1/n)
21
22    return r #Retorno de los numeros coprimos a n
23
24
25 print(euler_totient(5)) #Llamado de la funcion
26
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X]

4.0

Problema 5.


```
29  ## Problema5
30
31
32  def sec_euler_totient(n):
33
34      listc = []          # Creacion de la lista
35      for i in range (1, n+1): # Recorrido de las posiciones
36          if euler_totient(i) == i-1: # Comparacion entre posiciones
37              listc.append(i) # Se agrega el coprimo a la lista
38
39      return listc        # Se retorna la lista
40
41
42
43
44  print(euler_totient(10))
45  print(sec_euler_totient(10))
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

 Python Debug Console

4

[2, 3, 5, 7]

Problema 6

a. La funcion que permite obtener secuencias a partir de elementos de una lista se denomina **slice** y acepta los argumentos de inicio, fin y salto.

Sintaxis: `slice(start, end, step)`

b. La funcion que permite obtener las permutaciones de una lista en python se demonina **permutations** y acepta los siguientes argumentos:

iterable: hace mencion a una lista o tupla.

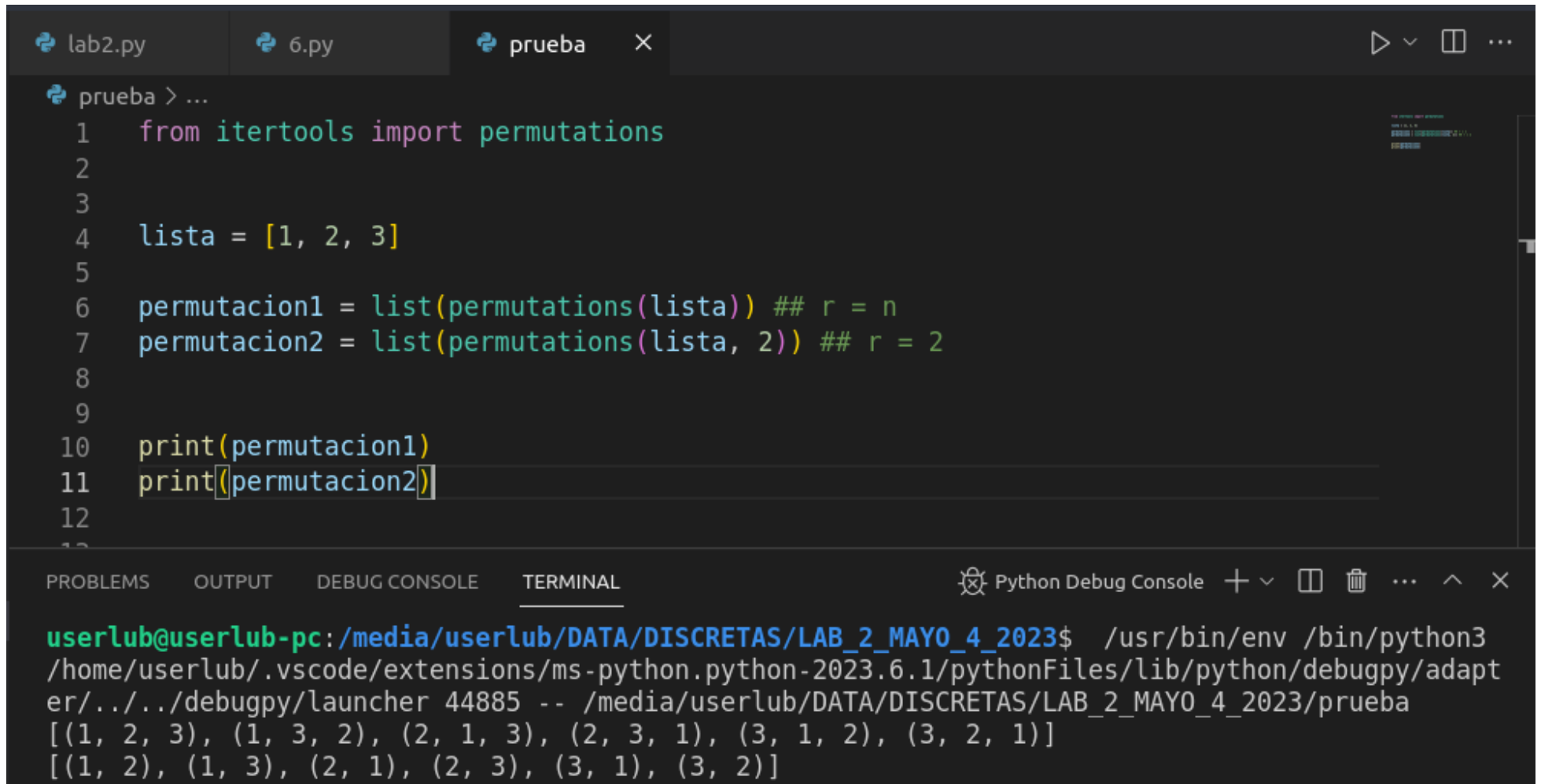
r: representa la longitud de cada permutacion generada. De no especificarse r sera igual a la longitud del iterable.

Sintaxis: `permutations(iterable, r)`

c. Las permutaciones se calculan mediante la formula $P = n!$, siempre y cuando $n=r$. En caso contrario la formula es la siguiente:

$${}_nP_r = \frac{n!}{(n-r)!}$$

Para ilustrar el procedimiento en python se comparte el siguiente codigo:



The image shows a VS Code editor window with three tabs: 'lab2.py', '6.py', and 'prueba'. The 'prueba' tab is active, displaying a Python script that uses the 'itertools.permutations' function. The script defines a list 'lista' with values [1, 2, 3], generates all permutations of this list ('permutacion1') and permutations of length 2 ('permutacion2'), and prints both. Below the editor, the 'TERMINAL' panel shows the command used to run the script and its output. The output displays two lists of tuples: the first list contains all 6 permutations of [1, 2, 3], and the second list contains all 6 permutations of length 2 from [1, 2, 3].

```
prueba > ...
1  from itertools import permutations
2
3
4  lista = [1, 2, 3]
5
6  permutacion1 = list(permutations(lista)) ## r = n
7  permutacion2 = list(permutations(lista, 2)) ## r = 2
8
9
10 print(permutacion1)
11 print(permutacion2)
12
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 44885 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/prueba
[(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)]
[(1, 2), (1, 3), (2, 1), (2, 3), (3, 1), (3, 2)]
```

d. La funcion que permite obtener las combinaciones de una lista en python se denomina **combinations** y acepta los siguientes argumentos:

iterable: hace mencion a una lista o tupla.

r: representa la longitud de cada permutacion generada. En las pruebas realizadas se pudo determinar que r es un argumento requerido.

Sintaxis: combinations(*iterable*, *r*)

e. Las combinaciones se calculan empleando la siguiente formula:

$${}^nC_r = \frac{n!}{(n-r)!r!}$$

Para ilustrar el procedimiento en python se comparte el siguiente codigo:

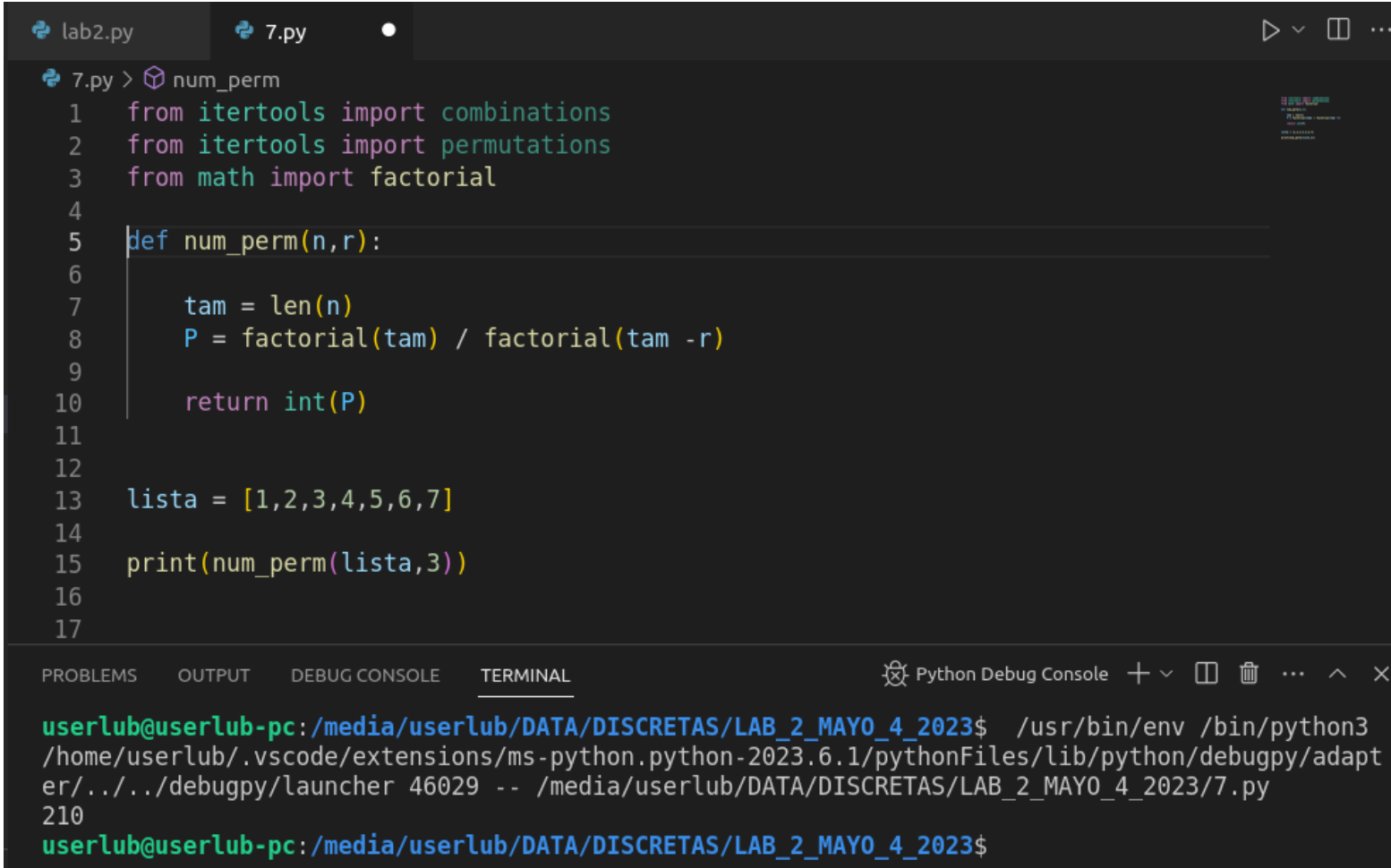
```
prueba > ...
1  from itertools import combinations
2
3
4  lista = ["a", "b", "c"]
5
6  combinacion1 = list(combinations(lista, 3)) ## r = n
7  combinacion2 = list(combinations(lista, 2)) ## r = 2
8
9
10 print(combinacion1)
11 print(combinacion2)
12
13
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 50167 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/prueba
[('a', 'b', 'c')]
[('a', 'b'), ('a', 'c'), ('b', 'c')]
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$
```

Problema 7.

a. Lista con 7 elementos, numero de permutaciones siendo $r = 3$



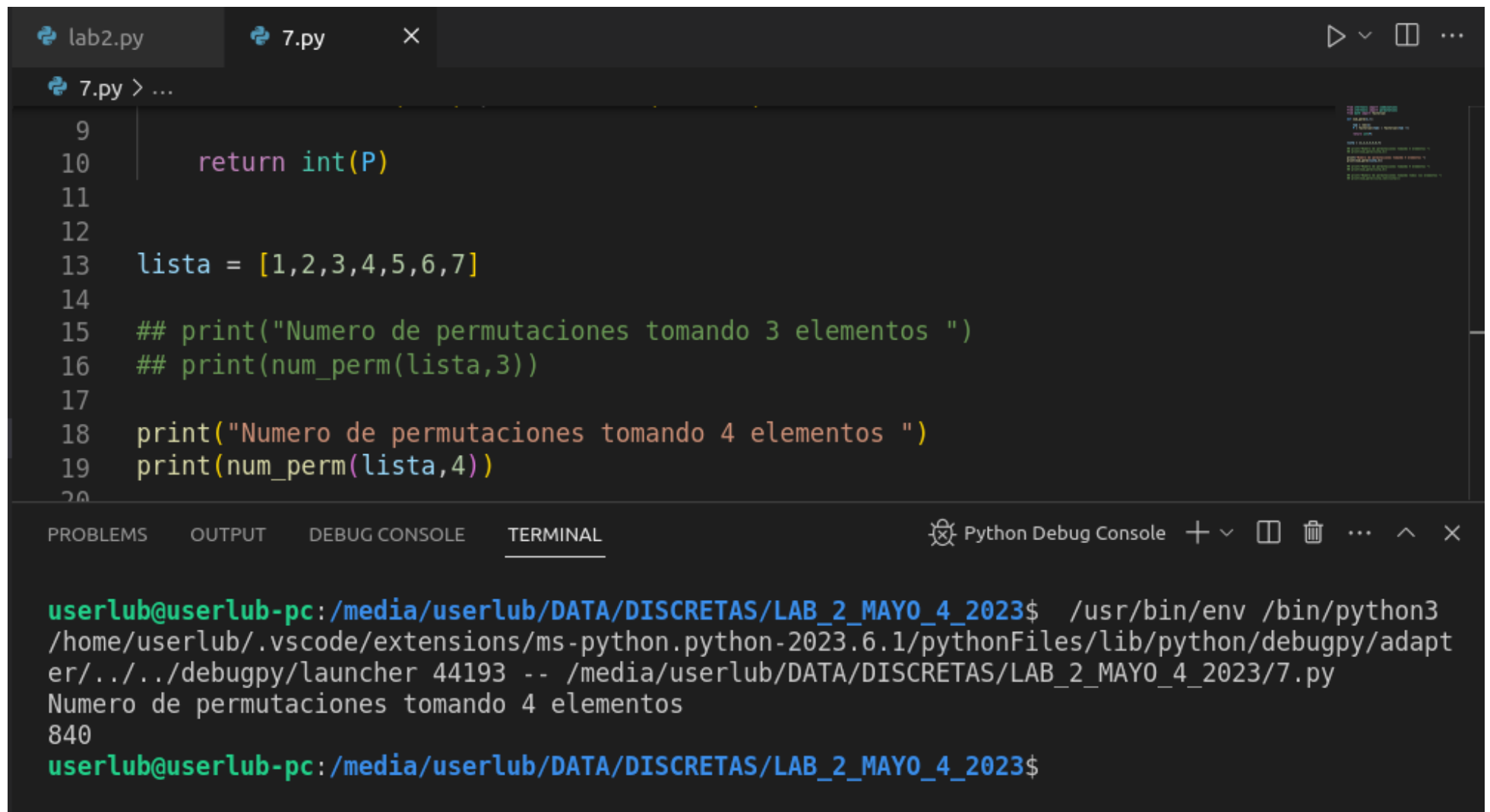
The image shows a VS Code editor window with a Python file named 7.py. The code defines a function num_perm(n, r) that calculates the number of permutations of n elements taken r at a time using the formula $P = \frac{n!}{(n-r)!}$. It then applies this function to a list [1, 2, 3, 4, 5, 6, 7] with r=3. Below the editor, a terminal window shows the command to run the script, which executes successfully.

```
lab2.py 7.py
7.py > num_perm
1 from itertools import combinations
2 from itertools import permutations
3 from math import factorial
4
5 def num_perm(n,r):
6
7     tam = len(n)
8     P = factorial(tam) / factorial(tam - r)
9
10    return int(P)
11
12
13 lista = [1,2,3,4,5,6,7]
14
15 print(num_perm(lista,3))
16
17
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [x] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 46029 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/7.py
210
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$
```

b. Lista con 7 elementos, numero de permutaciones siendo $r = 4$



The image shows a Visual Studio Code editor window with a Python file named `7.py` open. The code defines a function `num_perm` that calculates the number of permutations of a list of size `n` taken `r` at a time. The function uses a recursive approach, with a base case for `r = 1` returning the length of the list. The main part of the script defines a list `lista = [1, 2, 3, 4, 5, 6, 7]` and prints the results of `num_perm(lista, 3)` and `num_perm(lista, 4)`.

```
9
10     return int(P)
11
12
13 lista = [1,2,3,4,5,6,7]
14
15 ## print("Numero de permutaciones tomando 3 elementos ")
16 ## print(num_perm(lista,3))
17
18 print("Numero de permutaciones tomando 4 elementos ")
19 print(num_perm(lista,4))
20
```

The terminal output shows the command to run the script and the resulting output:

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 44193 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/7.py
Numero de permutaciones tomando 4 elementos
840
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$
```

c. Lista con 7 elementos, numero de permutaciones siendo $r = 5$

```

5  def num_perm(n,r):
6
7      tam = len(n)
8      P = factorial(tam) / factorial(tam - r)
9
10     return int(P)
11
12
13     lista = [1,2,3,4,5,6,7]
14
15     ## print("Numero de permutaciones tomando 3 elementos ")
16     ## print(num_perm(lista,3))
17
18     ## print("Numero de permutaciones tomando 4 elementos ")
19     ## print(num_perm(lista,4))
20
21     print("Numero de permutaciones tomando 5 elementos ")
22     print(num_perm(lista,5))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Python Debug Console + ▾ □ 🗑 ...

```

er/../../debugpy/launcher 53779 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/7.py
Numero de permutaciones tomando 5 elementos
2520

```

d. Lista con 7 elementos, numero de permutaciones siendo $r = n$

```
5 def num_perm(n,r):
6
7     tam = len(n)
8     P = factorial(tam) / factorial(tam - r)
9
10    return int(P)
11
12
13    lista = [1,2,3,4,5,6,7]
14
15    ## print("Numero de permutaciones tomando 3 elementos ")
16    ## print(num_perm(lista,3))
17
18    ## print("Numero de permutaciones tomando 4 elementos ")
19    ## print(num_perm(lista,4))
20
21    ## print("Numero de permutaciones tomando 5 elementos ")
22    ## print(num_perm(lista,5))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python De

userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/7.py
Numero de permutaciones tomando todos los elementos
5040

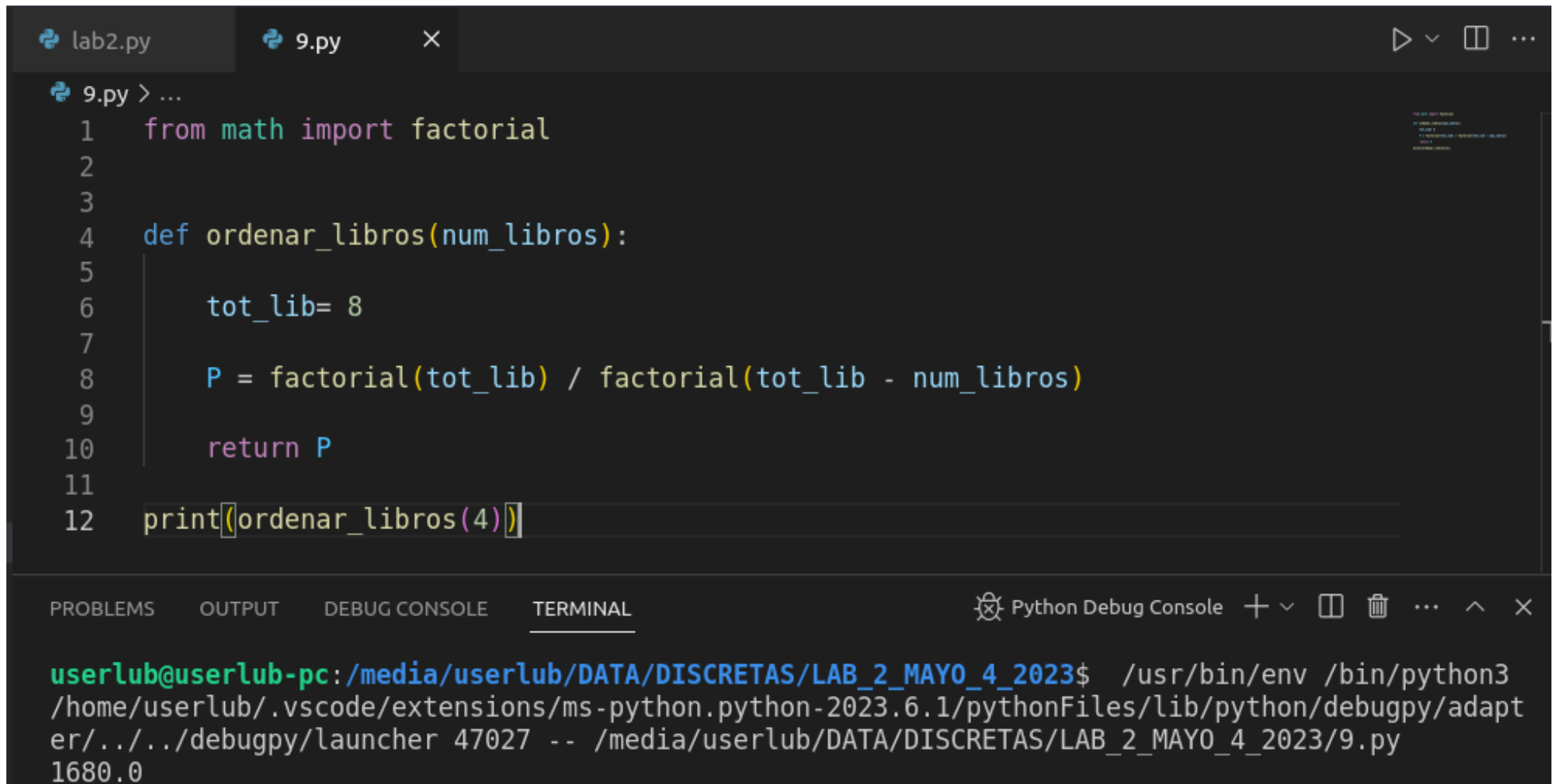
Problema 8.

```
8.py > ...
1 |
2 def lanzamiento_dado(num_lanz):
3 |
4     n = 8
5     S = pow(8,num_lanz)
6 |
7     return S
8 |
9 |
10 |
11 print(lanzamiento_dado(4))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 40621 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/8.py
4096
```

Problema 9



The image shows a VS Code editor window with two tabs: 'lab2.py' and '9.py'. The '9.py' tab is active, displaying a Python script. The script defines a function 'ordenar_libros' that calculates the number of permutations of 8 books taken 4 at a time. The script is as follows:

```
9.py > ...
1  from math import factorial
2
3
4  def ordenar_libros(num_libros):
5
6      tot_lib= 8
7
8      P = factorial(tot_lib) / factorial(tot_lib - num_libros)
9
10     return P
11
12  print(ordenar_libros(4))
```

Below the editor, the 'TERMINAL' panel is open, showing the command prompt and the output of the script:

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 47027 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/9.py
1680.0
```

Problema 10

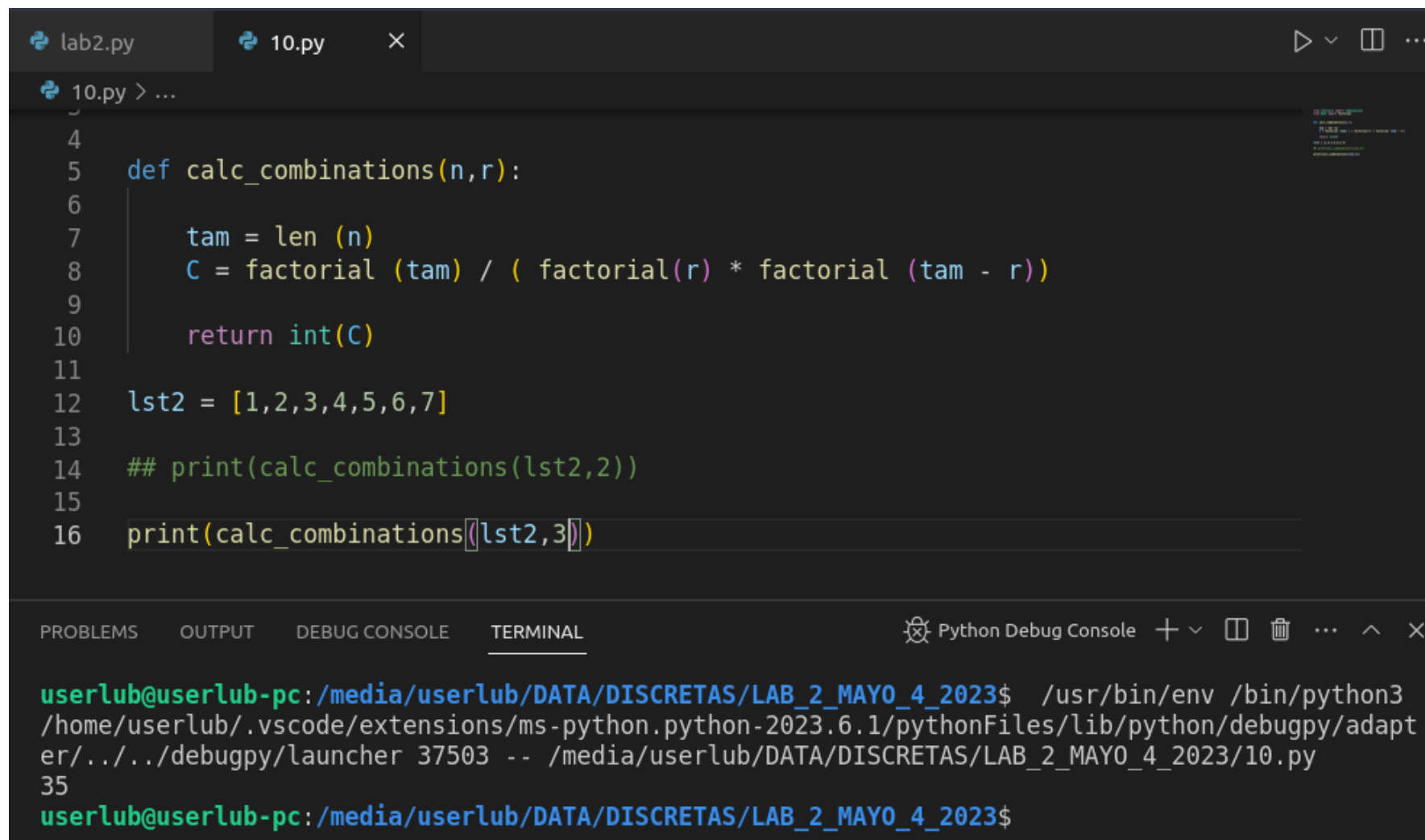
a. Lista con 7 elementos, combinacion donde $r = 2$

```
10.py > ...
1  from itertools import combinations
2  from math import factorial
3
4
5  def calc_combinations(n,r):
6
7      tam = len (n)
8      C = factorial (tam) / ( factorial(r) * factorial (tam - r))
9
10     return int(C)
11
12     lst2 = [1,2,3,4,5,6,7]
13
14     print(calc_combinations(lst2,2))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 38857 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/10.py
21
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$
```

b. Lista con 7 elementos, combinacion donde $r = 3$



The image shows a VS Code editor window with two tabs: 'lab2.py' and '10.py'. The '10.py' tab is active, displaying a Python script. The script defines a function 'calc_combinations(n,r)' that calculates the number of combinations using the formula $C = \frac{\text{factorial}(n)}{\text{factorial}(r) * \text{factorial}(n - r)}$. It then defines a list 'lst2 = [1,2,3,4,5,6,7]' and prints the result of 'calc_combinations(lst2,3)'. Below the editor, the 'TERMINAL' panel shows the command prompt output, indicating the script was executed successfully.

```
lab2.py 10.py ×  
10.py > ...  
4  
5 def calc_combinations(n,r):  
6  
7     tam = len (n)  
8     C = factorial (tam) / ( factorial(r) * factorial (tam - r))  
9  
10    return int(C)  
11  
12    lst2 = [1,2,3,4,5,6,7]  
13  
14    ## print(calc_combinations(lst2,2))  
15  
16    print(calc_combinations([lst2,3]))  
  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL  
Python Debug Console  
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3  
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt  
er/../../debugpy/launcher 37503 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/10.py  
35  
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$
```

c. Lista con 7 elementos, combinacion donde $r = 4$

```

5  def calc_combinations(n,r):
6
7      tam = len (n)
8      C = factorial (tam) / ( factorial(r) * factorial (tam - r))
9
10     return int(C)
11
12  lst2 = [1,2,3,4,5,6,7]
13
14  ## print(calc_combinations(lst2,2))
15
16  ## print(calc_combinations(lst2,3))
17
18  print(calc_combinations(lst2,4))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Python Debug Console + - [] [X] ... ^ X

```

userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 57001 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/10.py
35
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$

```

d. Lista con 7 elementos, combinacion donde $r = 5$

```

5  def calc_combinations(n,r):
6
7      tam = len (n)
8      C = factorial (tam) / ( factorial(r) * factorial (tam - r))
9
10     return int(C)
11
12     lst2 = [1,2,3,4,5,6,7]
13
14     ## print(calc_combinations(lst2,2))
15
16     ## print(calc_combinations(lst2,3))
17
18     ## print(calc_combinations(lst2,4))
19
20     print(calc_combinations(lst2,5))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Python Debug Console + - [] [X] ... ^ X

```

userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 40073 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/10.py
21
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$

```

Problema 11

```
prueba > ...  
9  
10 def imprimir_manos(num_cartas):  
11     palos = ['♠', '♥', '♦', '♣']  
12     rango = [str(n) for n in range(2,11)] + ['J','Q','K','A']  
13  
14     cartas = list(it.product(rango, palos))  
15  
16     manos = list(it.combinations(cartas, num_cartas)) # Combinaciones posibles  
17  
18     return len(manos) #longitud de la variable manos  
19  
20  
21  
22 print(imprimir_manos(6))  
23  
24  
25
```




PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python Debug Console + - [] [X] ... ^ X

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3  
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt  
er/../../debugpy/launcher 56993 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/prueba  
20358520
```

Problema 12

```
8
9 def lanzamiento_moneda(num_lanz):
10     caras = 2 * num_lanz # Total de caras en num_lanz
11     k = num_lanz
12     C = math.factorial(caras) / (math.factorial(k) * math.factorial(caras-k))
13     return int(C)
14
15
16
17 print(lanzamiento_moneda(5))
18
19
20
21
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

 Python Debug Console + ▾   ... ^

```
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$ /usr/bin/env /bin/python3
/home/userlub/.vscode/extensions/ms-python.python-2023.6.1/pythonFiles/lib/python/debugpy/adapt
er/../../debugpy/launcher 47251 -- /media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023/prueba
252
userlub@userlub-pc:/media/userlub/DATA/DISCRETAS/LAB_2_MAYO_4_2023$
```