

## Informe del Examen Final de Arquitectura de Computadora

### Integrantes:

Arturo Manuel Noriega Noriega 20190411

Carlos Sanchez Atiquipa 20141503

### 1) Resultados de la pregunta 1:

\_\_\_\_ Nota: El término potencia global se tomó como potencial global activa

- a. Está el script de la descarga del archivo household\_power\_consumption.csv

```
def get_csv():
    csv_url = 'https://jobenas-misc-bucket.s3.amazonaws.com/household_power_consumption.csv'
    req = requests.get(csv_url)
    url_content = req.content
    csv_name = csv_url.split('/')[3]
    csv_file = open(csv_name, 'wb')
    csv_file.write(url_content)
    csv_file.close()
```

- b. En la imagen, se muestra el script de ejecución de get\_cols() y la salida.

```
147 # Obtenemos los nombres de las columnas
148 cabecera = get_cols()
149 print(cabecera)
150
151 # fecha = '2006/12/16'
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2\$ python3 funciones.py  
datetime,Global\_active\_power,Global\_reactive\_power,Voltage,Global\_intensity,Sub\_metering\_1,Sub\_metering\_2,Sub\_metering\_3,sub\_metering\_4

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2\$

- c. En la imagen, se muestra la ejecución de get\_day(fecha) y la salida en el terminal

```
133
134 fecha = '2006/12/16'
135 # Obtenemos los datos de la fecha introducida
136 datos = get_day(fecha)
137 pp = pprint.PrettyPrinter(indent=4)
138 pp.pprint(datos)
139
140 # Obtenemos el promedio de potencia de la fecha introducida
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2\$ python3 funciones.py

```
( array([[4.2160000e+00, 4.1800000e-01, 2.3484000e+02, ..., 1.0000000e+00,
        1.7000000e+01, 5.2266670e+01],
        [5.3600000e+00, 4.3600000e-01, 2.3363000e+02, ..., 1.0000000e+00,
        1.6000000e+01, 7.2333336e+01],
        [5.3740000e+00, 4.9800000e-01, 2.3329000e+02, ..., 2.0000000e+00,
        1.7000000e+01, 7.0566666e+01],
        ...,
        [2.7340000e+00, 6.2000000e-02, 2.3981000e+02, ..., 2.0000000e+00,
        0.0000000e+00, 4.3566666e+01],
        [2.7640000e+00, 6.6000000e-02, 2.4127000e+02, ..., 1.0000000e+00,
        0.0000000e+00, 4.5066666e+01],
        [2.5860000e+00, 7.0000000e-02, 2.4163000e+02, ..., 1.0000000e+00,
        0.0000000e+00, 4.2100000e+01]]),
  [ '17:24:00',
    '17:25:00',
    '17:26:00',
    '17:27:00',
    '17:28:00',
```

- d. En la imagen, se muestra la ejecución de get\_mean() y la salida en el terminal

```
139 fecha = '2007/12/16'
140 # Obtenemos el promedio de potencia de la fecha introducida
141 promedio = get_mean(fecha)
142 print(f'Promedio de potencia global del dia {fecha}: {promedio}')
143
144 # Obtenemos la potencia maxima de la fecha introducida
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 funciones.py
Promedio de potencia global del dia 2007/12/16: 1.918
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$
```

- e. En la imagen, se muestra la ejecucion de `get_max()` y la salida en el terminal, donde se ve la potencia maxima de esa fecha y la hora que registró la potencia máxima.

```
144 fecha = '2007/12/16'
145 # Obtenemos la potencia maxima de la fecha introducida
146 potencia_max = get_max(fecha)
147 print(f'Potencia maxima global del dia {fecha}: {potencia_max}')
148
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 funciones.py
Potencia maxima global del dia 2007/12/16: {'potencia': 7.908, 'hora': '12:38:00'}
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$
```

- f. En la imagen, se muestra la ejecucion de `get_min()` y su salida en el terminal, donde se ve la potencia minima de la fecha de entrada y la hora que registró tal valor.

```
149 fecha = '2007/12/16'
150 # Obtenemos la potencia maxima de la fecha introducida
151 potencia_min = get_min(fecha)
152 print(f'Potencia minima global del dia {fecha}: {potencia_min}')
153
154 # Creamos el diccionario con fecha inicio y fecha fin
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 funciones.py
Potencia minima global del dia 2007/12/16: {'potencia': 0.216, 'hora': '07:26:00'}
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$
```

- g. En la imagen, se muestra la ejecucion de `get_day_dict` y su salida en el terminal, que ingresando un periodo de tiempo(entre 2 fechas) su salida es un diccionario cuya llave es la fecha y su valor es una lista de lista de valores de ese momento.

```

154 # Creamos el diccionario con fecha inicio y fecha fin
155 diccionario = gen_day_dict(['2008/08/21', '2008/08/22'])
156 pp = pprint.PrettyPrinter(indent=4)
157 pp.pprint(diccionario)

```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 funciones.py
{ '2008/08/21': array([[8.000000e-02, 0.000000e+00, 2.397800e+02, ..., 0.000000e+00,
0.000000e+00, 1.333333e+00],
[8.000000e-02, 0.000000e+00, 2.401600e+02, ..., 0.000000e+00,
1.000000e+00, 3.333337e-01],
[8.000000e-02, 0.000000e+00, 2.401300e+02, ..., 0.000000e+00,
1.000000e+00, 3.333337e-01],
...,
[8.200000e-02, 0.000000e+00, 2.409900e+02, ..., 0.000000e+00,
0.000000e+00, 1.366667e+00],
[8.200000e-02, 0.000000e+00, 2.409900e+02, ..., 0.000000e+00,
1.000000e+00, 3.666667e-01],
[8.200000e-02, 0.000000e+00, 2.408600e+02, ..., 0.000000e+00,
1.000000e+00, 3.666667e-01]]),
'2008/08/22': array([[8.200000e-02, 0.000000e+00, 2.408700e+02, ..., 0.000000e+00,
0.000000e+00, 1.366667e+00],
[1.240000e-01, 5.600000e-02, 2.411100e+02, ..., 0.000000e+00,
1.000000e+00, 1.066666e+00],
[2.100000e-01, 1.720000e-01, 2.412600e+02, ..., 0.000000e+00,
1.000000e+00, 2.500000e+00],
...,
[2.600000e-01, 2.620000e-01, 2.427000e+02, ..., 0.000000e+00,
1.000000e+00, 3.333335e+00],
[2.580000e-01, 2.620000e-01, 2.427800e+02, ..., 0.000000e+00,
1.000000e+00, 3.300002e+00],
[2.560000e-01, 2.620000e-01, 2.429300e+02, ..., 0.000000e+00,
0.000000e+00, 4.266670e+00]])}
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$

```

- h. En la imagen, se muestra la ejecución de `get_exec_time_a`, que toma el tiempo de ejecución de las operaciones de descarga del archivo `hosuehold_power_consumption.csv` y el tiempo de ejecución es de 142621364.8 us o 142.6 segundos

```

165 if __name__ == "__main__":
166     get_exec_time_a()
167     #get_exec_time_b()

```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El tiempo de ejecución para leer el archivo CSV es: 142621364.8us
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$

```

- i. En la imagen, se visualiza la ejecución de la función `get_exec_time_b()`, cuya salida es el tiempo de ejecución de la función `get_day_dict` entre las fechas de 2008/08/21 y 2008/08/30

```

165 if __name__ == "__main__":
166     #get_exec_time_a()
167     get_exec_time_b()
168     #calc_speedup_e()

```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El tiempo de ejecución para obtener los días en el archivo CSV es: 63350957.5us
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El tiempo de ejecución para obtener los días en el archivo CSV es: 82366790.8us
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El tiempo de ejecución para obtener los días en el archivo CSV es: 64768034.4us
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$

```

- j. En la imagen, se visualiza la ejecución de la función `gen_day_dict_thread`, que usa hilos para extraer los datos referido a fechas y guardarla en un diccionario cuya llave es la fecha.

```

168     diccionario=gen_day_dict_threaded(['2008/08/21','2008/08/22'])
169     print(diccionario)
170
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
{'2008/08/21': array([[8.0000000e-02, 0.0000000e+00, 2.3978000e+02, ..., 0.0000000e+00,
0.0000000e+00, 1.3333334e+00],
[8.0000000e-02, 0.0000000e+00, 2.4016000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3333337e-01],
[8.0000000e-02, 0.0000000e+00, 2.4013000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3333337e-01],
...,
[8.2000000e-02, 0.0000000e+00, 2.4099000e+02, ..., 0.0000000e+00,
0.0000000e+00, 1.3666667e+00],
[8.2000000e-02, 0.0000000e+00, 2.4099000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.6666667e-01],
[8.2000000e-02, 0.0000000e+00, 2.4086000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.6666667e-01]]), '2008/08/22': array([[8.2000000e-02, 0.0000000e+00, 2.4087000e+02, ..., 0.0000000e+00,
0.0000000e+00, 1.3666667e+00],
[1.2400000e-01, 5.6000000e-02, 2.4111000e+02, ..., 0.0000000e+00,
1.0000000e+00, 1.0666666e+00],
[2.1000000e-01, 1.7200000e-01, 2.4126000e+02, ..., 0.0000000e+00,
1.0000000e+00, 2.5000000e+00],
...,
[2.6000000e-01, 2.6200000e-01, 2.4270000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3333335e+00],
[2.5800000e-01, 2.6200000e-01, 2.4278000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3000002e+00],
[2.5600000e-01, 2.6200000e-01, 2.4293000e+02, ..., 0.0000000e+00,
0.0000000e+00, 4.2666670e+00]]])}
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$

```

- k. Al igual que en punto j, se muestra en la siguiente imagen la ejecución de la función `gen_day_dict_thread`, que usa multiprocesos para extraer los datos.

```

172     numproc=input("Ingrese el numero de procesos: ")
173     numproc=int(numproc)
174     diccionario=gen_day_dict_multi('2008/08/21','2008/08/22',numproc)
175     print(diccionario)
176
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 2
{'2008/08/21': array([[8.0000000e-02, 0.0000000e+00, 2.3978000e+02, ..., 0.0000000e+00,
0.0000000e+00, 1.3333334e+00],
[8.0000000e-02, 0.0000000e+00, 2.4016000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3333337e-01],
[8.0000000e-02, 0.0000000e+00, 2.4013000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3333337e-01],
...,
[8.2000000e-02, 0.0000000e+00, 2.4099000e+02, ..., 0.0000000e+00,
0.0000000e+00, 1.3666667e+00],
[8.2000000e-02, 0.0000000e+00, 2.4099000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.6666667e-01],
[8.2000000e-02, 0.0000000e+00, 2.4086000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.6666667e-01]]), '2008/08/22': array([[8.2000000e-02, 0.0000000e+00, 2.4087000e+02, ..., 0.0000000e+00,
0.0000000e+00, 1.3666667e+00],
[1.2400000e-01, 5.6000000e-02, 2.4111000e+02, ..., 0.0000000e+00,
1.0000000e+00, 1.0666666e+00],
[2.1000000e-01, 1.7200000e-01, 2.4126000e+02, ..., 0.0000000e+00,
1.0000000e+00, 2.5000000e+00],
...,
[2.6000000e-01, 2.6200000e-01, 2.4270000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3333335e+00],
[2.5800000e-01, 2.6200000e-01, 2.4278000e+02, ..., 0.0000000e+00,
1.0000000e+00, 3.3000002e+00],
[2.5600000e-01, 2.6200000e-01, 2.4293000e+02, ..., 0.0000000e+00,
0.0000000e+00, 4.2666670e+00]]])}
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$

```

- l. En la imagen, se muestra el calculo del speedup de la función `gen_day_dict` ('2008/08/21','2007/09/01') para 2, 4 y 8 procesos , y se puede visualizar que con threads, el speedup es menor a 1 y , en multiprocessing, aumenta. Por lo que , el proceso en paralelo con thread aplicado no es conveniente. Los resultados de las 3 funciones se verifican con la funcion `verif_gen_day_dict`

```

182     calc_speedup_e()
183     #verif_func_gen()
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 2
El speed_up con threading: 0.898
El speed_up con multiprocessing: 1.424
El mejor desempeño la tuvo el multiprocessing
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 4
El speed_up con threading: 0.905
El speed_up con multiprocessing: 1.788
El mejor desempeño la tuvo el multiprocessing
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 8
El speed_up con threading: 0.878
El speed_up con multiprocessing: 1.887
El mejor desempeño la tuvo el multiprocessing
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$

```

- m. En la imagen, se muestra el resultado de la función `calc_stats` que comprende esas dos fechas, y cuyo resultado es la potencia activa máxima, mínima, promedio y total, guardada en un diccionario.

```
185     diccionario={}
186     diccionario=calc_stats('2008/08/21','2008/08/30')
187     print(diccionario)
188     #diccionario=calc_stats_conc('2008/08/21','2008/08/22')

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
{'2008/08/21': {'potencia_Maxima': 0.926, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.181, 'Total': 261.298}, '2008/08/22': {'potencia_Maxima': 1.02, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.177, 'Total': 254.642}, '2008/08/24': {'potencia_Maxima': 0.912, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.177, 'Total': 254.716}, '2008/08/25': {'potencia_Maxima': 0.958, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.174, 'Total': 250.298}, '2008/08/26': {'potencia_Maxima': 0.952, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.179, 'Total': 257.564}, '2008/08/27': {'potencia_Maxima': 1.006, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.179, 'Total': 258.114}, '2008/08/28': {'potencia_Maxima': 1.18, 'potencia_Minima': 0.076, 'potencia_Promedio': 0.186, 'Total': 268.246}, '2008/08/29': {'potencia_Maxima': 0.924, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.188, 'Total': 270.424}}
```

- n. En la imagen se muestra el resultado de la función `calc_stats_conc`, y `calc_stats` son iguales.

```
187     diccionario={}
188     #diccionario=calc_stats('2008/08/21','2008/08/30')
189     #print(diccionario)
190     diccionario=calc_stats_conc('2008/08/21','2008/08/30')
191     print(diccionario)

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
{'2008/08/21': {'potencia_Maxima': 0.926, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.181, 'Total': 261.298}, '2008/08/22': {'potencia_Maxima': 1.02, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.177, 'Total': 254.642}, '2008/08/24': {'potencia_Maxima': 0.912, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.177, 'Total': 254.716}, '2008/08/25': {'potencia_Maxima': 0.958, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.174, 'Total': 250.298}, '2008/08/26': {'potencia_Maxima': 0.952, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.179, 'Total': 257.564}, '2008/08/27': {'potencia_Maxima': 1.006, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.179, 'Total': 258.114}, '2008/08/28': {'potencia_Maxima': 1.18, 'potencia_Minima': 0.076, 'potencia_Promedio': 0.186, 'Total': 268.246}, '2008/08/29': {'potencia_Maxima': 0.924, 'potencia_Minima': 0.078, 'potencia_Promedio': 0.188, 'Total': 270.424}}
```

- n. En la primera imagen, se muestra el `calc_speedup_h()` con `calc_stats_conc` usando threads, y los resultados de speedup es menor que uno, mientras en la segunda imagen se usa `multiprocess` variando la cantidad de procesos y se muestra que aumentando la cantidad de procesos, aumenta el speedup entre el tiempo de ejecución de `calc_stats` y `calc_stats_conc`. Finalmente, la concurrencia adecuada para esta función es mediante `multiprocessing`. Se puede verificar los resultados de ambas funciones mediante la función `verif_calc_stats()`.

```
202     calc_speedup_h()
203     #verif_calc_stats()

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El speed_up con threading para calc_stats: 0.906
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El speed_up con threading para calc_stats: 0.9
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
El speed_up con threading para calc_stats: 0.857
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$
```

```
205     calc_speedup_h()
206     #verif_calc_stats()

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 2
El speed_up con multiprocessing para calc_stats: 1.725
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 4
El speed_up con multiprocessing para calc_stats: 2.653
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$ python3 preg2.py
Ingrese el numero de procesos: 8
El speed_up con multiprocessing para calc_stats: 3.959
carlossa@DESKTOP-NUPRJ84:/mnt/c/Users/Acer/desktop/mandar/EX2-2021-1/ex2$
```

## Conclusiones:

Debido a los resultados obtenidos, la implementación con `threading` para este caso no es la adecuada por lo que se recomienda considerar los `multiprocessing` para las funciones.

Una posible causa de que el speed up sea menor a 1 con `threading` se debe a que los intervalos considerados son pequeños; por lo que con un mayor intervalo posiblemente se note una mejora.

## 2) Resultados de la pregunta2:

En el scrip de python se realizaron 2 hilos para permitir que se conectaran de manera secuencial los dos jugadores. A continuación se observa la confirmación del ingreso a la sala de ambos jugadores conjuntamente con la tabla de cada jugador.

Se evidencia el inciso a , b y c.

Nota: primero ejecutar ambos programas , luego solamente escribir en el jugador número 1 el nombre de sala 1 y luego completar escoger la opción del tablero. Finalmente solo ingresar en el jugador número 2 el nombre de la sala.

<pre>PS D:\Python3\Pregrunta2_Examen_final&gt; py server_Battelship.py Valor de SOCK BUFFER: 4 Esperando conexiones Esperando conexiones Cliente cerro la conexio Cliente cerro la conexio []</pre>	<pre>PS D:\Python3\Pregrunta2_Examen_final&gt; py cliente1_Battelship.py Usted es el jugador 1 Ingrese el nombre de la sala: sala1 Ingrese su opción de tablero = opción "1" de 5 x5, "2" de 8 x 8 y "3 de 10 x 10: 1 00000 00000 00000 00000 00000  Cierro conexion PS D:\Python3\Pregrunta2_Examen_final&gt; []</pre>	<pre>PS D:\Python3\Pregrunta2_Examen_final&gt; py cliente1_Battelship.py Usted es el jugador 2 Ingrese el nombre de la sala: sala1 Usted ha ingresado a la sala 00000 00000 00000 00000 00000  Cierro conexion PS D:\Python3\Pregrunta2_Examen_final&gt; []</pre>	<div><div>py</div><div>powershell</div><div>powershell</div></div>
---	---	---	--

Ve a Configuración para activar Windows.

### Referencias:

El código battleship\_solution.py con url

<https://paideia.pucp.edu.pe/cursos/mod/resource/view.php?id=1929220>

Módulo threading usado

[https://www.tutorialspoint.com/python/python\\_multithreading.htm](https://www.tutorialspoint.com/python/python_multithreading.htm)

Módulo pool

<https://docs.python.org/3/library/concurrent.futures.html#>