

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ARQUITECTURA DE COMPUTADORAS

Laboratorio 9

(2023-1)

Indicaciones generales:

- Materiales permitidos: Wiki del curso, apuntes de clase o documentación de python online.
- **Está prohibido el uso de cualquier modelo de lenguaje** como ChatGPT o Github Copilot. A cualquier alumno que se le detecte que ha consultado un modelo de lenguaje se le pondrá nota 0 (cero) en el laboratorio.
- **No está permitido el uso de la librería pandas**
- Usted debe subir a Paideia 1 solo archivo comprimido(.zip o .rar) con el nombre L9_CODIGOPUCP.zip o L9_CODIGOPUCP.rar. Este archivo comprimido debe tener archivos de python para cada pregunta. **No se aceptarán soluciones en jupyter notebook o pdf.**
- El horario máximo permitido para subir el archivo es a las 10:00:00 pm. Pasada esa hora, habrá una penalidad de 2 puntos por cada minuto extra que se demore en entregar su archivo.

Puntaje total: 20 puntos

Pregunta 1

Para esta pregunta se tiene el siguiente archivo pregunta1.txt. Se le solicita:

- Crear un **servidorMaster.py** el cual debe ejecutarse una única vez en todo el procedimiento de esta pregunta. Solo se conectará a un cliente al cual se le enviará la información de una fila del archivo .txt cada 1 segundo. Tenga en cuenta que si el cliente se desconecta, el servidor se quedará a la escucha hasta que se vuelva a conectar nuevamente un cliente y continuará enviando la información de la siguiente fila. No se reinicia, y si llegara el caso que llegue al final del archivo, este debe enviar nuevamente desde el principio. **(2.5 puntos)**
- Crear un **servidor.py** que se encargue de conectarse al **servidorMaster.py** y que reciba los datos que este le envía. Establezca un tamaño de buffer de 1024. Asimismo, por cada cadena recibida debe realizar el conteo de la cantidad de bytes, luego imprimir la cantidad y dicha cadena de la forma:

```
>>HOLA SOY SERVIDOR Y HE RECIBIDO DE MASTER 10 bytes: "#####"
```


Notar que la cadena recibida por **servidorMaste.py** es "#####" y la cantidad de bytes es 10. **(3.5 puntos)**
- En el mismo archivo de **servidor.py** debe establecer conexión con un cliente para poder enviar la cantidad de caracteres previamente hallados. Solo debe enviar el número. **(2 puntos)**

- d. Crear un **cliente.py** que se encargue de conectarse al **servidor.py** y que reciba los datos que este le envía. Este archivo solo se encargará de imprimir la cantidad de caracteres contados por servidor de la siguiente forma:

```
>>HOLA SOY CLIENTE Y LA CANTIDAD DE CARACTERES RECIBIDA POR
SERVIDOR ES: 10 (2.5 puntos)
```

- e. Con la herramienta `sudo tcpdump -i <interface> port <port_number>`. Analice el puerto que ha designado para **servidorMaster.py** y visualice si el tamaño de los bytes enviados corresponde a lo que imprimió en terminal en el inciso b. Adjunte una imagen (tcpdump.png) donde se observe la salida usando *tcpdump* y otra imagen (servidor.png) donde se observe el terminal en el que se ejecuta **servidor.py**. (1.5 punto)

```
102.785197 IP localhost.40064 > localhost.8083: Flags [P], ack 2350, win 511, options [nop,nop,TS val 3248630869 ecr 3248630869], length 0
103.786305 IP localhost.8083 > localhost.40064: Flags [P], seq 2350:2400, ack 1, win 512, options [nop,nop,TS val 3248631871 ecr 3248630869], length 50
103.786319 IP localhost.40064 > localhost.8083: Flags [F], ack 2400, win 511, options [nop,nop,TS val 3248631871 ecr 3248631871], length 0
104.787399 IP localhost.8083 > localhost.40064: Flags [P], seq 2400:2450, ack 1, win 512, options [nop,nop,TS val 3248632872 ecr 3248631871], length 50
104.787413 IP localhost.40064 > localhost.8083: Flags [F], ack 2450, win 511, options [nop,nop,TS val 3248632872 ecr 3248632872], length 0
105.788602 IP localhost.8083 > localhost.40064: Flags [P], seq 2450:2500, ack 1, win 512, options [nop,nop,TS val 3248633873 ecr 3248632872], length 50
105.788616 IP localhost.40064 > localhost.8083: Flags [F], ack 2500, win 511, options [nop,nop,TS val 3248633873 ecr 3248633873], length 0
106.789696 IP localhost.8083 > localhost.40064: Flags [P], seq 2500:2550, ack 1, win 512, options [nop,nop,TS val 3248634874 ecr 3248633873], length 50
106.789710 IP localhost.40064 > localhost.8083: Flags [F], ack 2550, win 511, options [nop,nop,TS val 3248634874 ecr 3248634874], length 0
107.790797 IP localhost.8083 > localhost.40064: Flags [P], seq 2550:2600, ack 1, win 512, options [nop,nop,TS val 3248635875 ecr 3248634874], length 50
107.790812 IP localhost.40064 > localhost.8083: Flags [F], ack 2600, win 511, options [nop,nop,TS val 3248635875 ecr 3248635875], length 0
108.791942 IP localhost.8083 > localhost.40064: Flags [P], seq 2600:2650, ack 1, win 512, options [nop,nop,TS val 3248636876 ecr 3248635875], length 50
108.791973 IP localhost.40064 > localhost.8083: Flags [F], ack 2650, win 511, options [nop,nop,TS val 3248636876 ecr 3248636876], length 0
109.792144 IP localhost.8083 > localhost.40064: Flags [P], seq 2650:2700, ack 1, win 512, options [nop,nop,TS val 3248637876 ecr 3248636876], length 50
109.792159 IP localhost.40064 > localhost.8083: Flags [F], ack 2700, win 511, options [nop,nop,TS val 3248637876 ecr 3248637876], length 0
110.793277 IP localhost.8083 > localhost.40064: Flags [P], seq 2700:2750, ack 1, win 512, options [nop,nop,TS val 3248638878 ecr 3248637876], length 50
110.793292 IP localhost.40064 > localhost.8083: Flags [F], ack 2750, win 511, options [nop,nop,TS val 3248638878 ecr 3248638878], length 0
```

Nota el valor de `<interface>` usualmente es “lo”. Ejemplo:

```
$sudo tcpdump -i lo port 8081
```

Pregunta 2

Se tiene a disposición 50 archivos de tipos .csv de los cuales cada uno contiene 80000 filas y 8 columnas, donde cada celda tiene asignado un número aleatorio del 0 al 9.

- a. Desarrolle el programa **preg2_a.py**, en el cual con su código de alumno se le solicita hallar la cantidad de veces que este se encuentra en los archivos de forma horizontal. Es decir, debe revisar en cada fila de cada uno de los archivos si su código hace “match”. Luego debe mostrar un histograma que indique la cantidad de veces que hubo match en cada archivo, donde el eje X corresponderá a cada archivo y el eje Y a la cantidad de “matches”. Este histograma debe guardarse como imagen de la forma “**hist2_a.png**” (4 puntos)
- b. Desarrolle el programa **preg2_b.py**, y de forma análoga, debe realizar el mismo procedimiento pero para la columnas, tenga en cuenta que en una sola columna solo debe revisar de 8 en 8, es decir, que en una columna de N números, debe sólo realizar un análisis N/8 veces. Asimismo, el histograma debe tener el formato “**hist2_b.png**”. (3 puntos)
- c. Para el inciso (a) y (b) evalúe los tiempos de ejecución, de forma separada. Es decir, edite cada uno de los códigos para que se imprima en terminal el tiempo de ejecución del programa. Comente acerca de sus resultados (1 puntos)