



*Este examen consta de 2 ejercicios con un total de 40 puntos.*

Apellidos: \_\_\_\_\_ Nombre: \_\_\_\_\_ Grupo: \_\_\_\_\_

1. (20p) Se trata de construir un servidor proxy utilizando el protocolo SOCKS versión 4 descrito en la documentación adjunta.

El proxy aceptará todas las operaciones que se le soliciten, no es necesario dar soporte a ningún mecanismo de autenticación.

Se pide:

- Realizar un análisis adecuado del problema.
- Diseñar una solución para acometer el objetivo propuesto.
- Realizar una especificación en pseudocódigo, con especial énfasis en la parte de comunicaciones.
- Realizar un esbozo de implementación en lenguaje Python.
- Indicar qué argumentos habría que pasar al programa por línea de comandos y cuál sería la sintaxis de llamada del programa.

2. (20p) Se trata de programar una aplicación similar a netcat. El objetivo principal del programa es leer un flujo de bytes de su entrada estándar y enviarlos a través de un socket UDP, y de forma simultánea, leer los datos procedentes de socket y enviarlo a su salida estándar. La aplicación debe poder manejar tanto conexiones activas como pasivas. Como requisito obligatorio se prohíbe el uso de la llamada al sistema `select()`.

La sintaxis de llamada para el modo pasivo es:

```
netcat.py -u -l port
```

donde:

- **port:** puerto en el que escucha el programa.

La sintaxis de llamada para el modo activo:

```
netcat.py -u host port
```

donde:

- **host:** máquina donde reside el servidor.
- **port:** puerto del servidor.

Se pide implementación funcional completa en lenguaje Python.