

Examen extraordinaria

En este examen se evalúan los contenidos mínimos del módulo de PSP del ciclo DAM.

Consta de 3 ejercicios.

- 1º ejercicio 3 puntos.
- 2º ejercicio 3 puntos.
- 3º ejercicio 4 puntos.

Ejercicio 1 - Comunicación UDP.

Desarrolla el cliente y el servidor (MONOTHREAD).

El servicio contará las vocales, las consonantes y otros caracteres.

Ejemplo:

Cliente envía "Hola mundo!"
Servidor contesta con "4:5:2"

Siendo el primer número el número de vocales, el segundo el número de consonantes y el último el resto de caracteres. Da igual minúsculas y mayúsculas.

El cliente recibe por parámetro ip, puerto y mensaje.

Ejemplo:

- java examen.Ej1 192.168.1.1 8888 "Hola mundo!"

Respuesta de server -> 4:5:2

El servidor recibe por parámetro el puerto de escucha.

Ejemplo:

- java examen.Ej1Server 8888

Puntos:

- 1 punto comunicación cliente-servidor UDP
- 1 punto procesado de la información en el servidor
- 1 punto escribir respuesta correcta en el cliente

Ejercicio 2 - Servicio web

Crea un servidor multithread TCP que reciba peticiones HTTP y genere respuestas HTTP. Como cliente usarás un navegador. El servicio valida la letra de los DNI. En el navegador escribes:

http://localhost:8888/12345678Z
o
http://localhost:8888/00000000T

El servidor genera una respuesta HTTP con el texto OK dentro de un h1. En cambio si es erróneo genera una respuesta KO dentro de un h1 pero con color rojo.

Módulos DNI

RESTO	0	1	2	3	4	5	6	7	8	9	10	11
LETRA	T	R	W	A	G	M	Y	F	P	D	X	B

RESTO	12	13	14	15	16	17	18	19	20	21	22
LETRA	N	J	Z	S	Q	V	H	L	C	K	E

Puntos:

- 1 punto recibir DNI.
- 0.5 punto calculo letra.
- 0.5 punto generar respuesta ok.
- 1 punto generar ko con color.

Ejercicio 3 - Threads

Tienes un array de 32000 números enteros aleatorios entre 0 y 9

Genera unos threads que consultan 3 posiciones aleatorias de este array, suman los valores y verifican si ese número es primo. Cada thread hace esta acción hasta que consigue encontrar 10 número primos. El thread guarda un histórico de qué tres números ha consultado en un ArrayList de Strings, cada String son las posiciones separadas por ‘:’.

Ejemplo posición:

3435:6523:5

El thread principal genera un array de 3 de estos threads, los ejecuta y espera que terminen (Es indiferente el orden en el que terminan, hasta que no acaban todos, el thread principal no continua)

Cuando han terminado, el thread principal recoge los resultados y escribe las 30 combinaciones resultantes.

Puntos:

- 2 punto threads.
- 1 punto comienzo de threads y finalización.
- 1 punto mostrar resultado de la operación.