

Exercicio 1: IPS ON

A empresa de seguridade informática SegurNosa desexa unha aplicación que lle permita saber se existe algún equipo conectado nun rango de IPS. Para elo pensan que a solución podería ser realizar un programa que realice ping as diferentes IPS e recoller o resultado da execución do comando da seguinte maneira:

IPs ON

IP: 192.168.1.2 → Paquetes perdidos: 0% ; TTL: 128

IP: 192.168.1.3 → Paquetes perdidos: 10% ;TTL: 128

IPs OFF ou ERROR

IP: 192.168.1.1

IP: 192.168.1.4

ESTADÍSTICAS

1) *O 50% das IPS están ON*

2) *Tempo medio TTL: 128*

Indícanos que na primeira versión desexan eles poder introducir os dous primeiros parámetros do rango IP e logo que os seguintes sexan calculados, por exemplo: 192.168.XXX.YYY , donde XXX e YYY será calculado polo noso aplicativo.

Para resolver este problema será necesario:

- Crear un enumerado **StatusPing** con seguintes valores:
 - ON
 - ERROR
 - OFF
- Crear unha clase **IpV4** cos seguintes atributos:
 - Integer: atr1 (parámetro que introduce o usuario)
 - Integer: atr2 (parámetro que introduce o usuario)

- Integer: atr3
- Integer: atr4
- Status: ON / OFF / ERROR
- Float: ttl (resposta do comando ping)
- Tamén posuirá para cada atributo métodos para pasar a seguinte ip ou para retroceder de ip, isto quere dicir que o método nos retornará o valor. A nomenclatura ten que ser: incrementaAtr1() e decrementaAtr1;
- Clase **ProcesoPing**
Incluirá a lóxica de creación do proceso e recollida da información que retorne o comando PING.
- Clase **ProcesarSalidaPing**
 - Inclúe os seguintes atributos estáticos
 - Private List<IPv4> listaIPS , que conterá a información das IPS procesadas.
 - Inclúe un método estático que será parsearSalidaPing que teña como parámetro un obxecto IPv4 e un string do resultado do comando ping. Este método retorna void e actualiza a lista estática.
- Clase **MainPractica**
Terá o menor código posible permitindo nun futuro ter un selector de diferentes opcións.

Para resolver este problema debes empregar algunha das clases vistas para xestionar procesos, terase en conta que o código cumpra as convencións, teña comentarios apropiados e se documenten os diferentes métodos tanto entradas como saídas, posúa algoritmos claros, empregue as librerías de forma correcta.

Exercicio 2: IPS ON E RUTA

A mesma empresa solicita unha nova funcionalidade que consiste en que se inclúa a ruta que seguen os paquetes para chegar a IP destino, é dicir deberemos ampliar a funcionalidade do exercicio incorporando esta nova chamada. (Windows tracert)

Nota: O proxecto débese chamar PSPprc01_NOME_APELIDO.