

---

ESCOM-IPN

# Práctica 1

## Calculadora de IP

REDES DE COMPUTADORAS

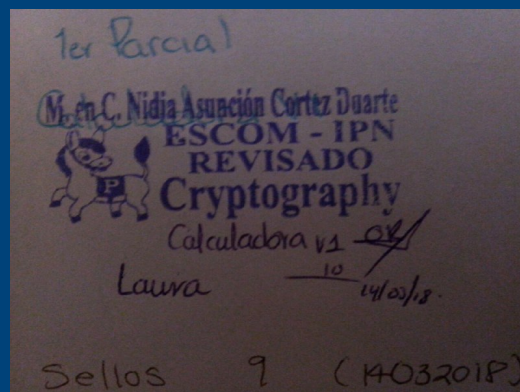
Laura Andrea Morales López

MSc. NIDIA ASUNCIÓN CORTEZ DUARTE

Marzo 2018

### Resumen

*In this report we will see some of the characteristics of the IP and provide a program that will calculate the IP Class, Range of host, Network and Broadcast.*



# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Marco teórico</b>	<b>2</b>
2.0.1. Problemas con IPv4 . . . . .	2
<b>3. Software</b>	<b>3</b>
<b>4. Procedimiento</b>	<b>3</b>
<b>5. Resultados</b>	<b>3</b>
<b>6. Discusión</b>	<b>5</b>
<b>7. Conclusión</b>	<b>5</b>
<b>8. Código</b>	<b>5</b>
<b>9. Mapa de memoria</b>	<b>7</b>

## 1. Introducción

## 2. Marco teórico

Debido a la cantidad de cables necesarios para conectar cada red con cada otra red del mundo no todas las redes tienen una conexión directa, es decir, no existe un cable entre tu red local y los servidores de Facebook por ejemplo.

Por eso existe el Protocolo IP que nos permite comunicarnos entre redes.

En resumen lo que permite es que tu red local solo esté conectada a unas pocas redes y a varios routers, estos tienen algo llamado una tabla de direcciones, que les permite navegar entre redes hasta encontrar su destino.

El enrutamiento es parecido a la recursión, en el sentido en que no soluciona tu problema sino que solo te lleva un paso más cerca.

Cuando conectamos un dispositivo (Computadora, Smartphone, etc) a una Red se genera una dirección diferente para cada uno de esos. Estas direcciones las llamamos IP.

Como fue originalmente desarrollado este esquema podría alojar un identificador de **32 bits** a cada dispositivo que se quisiera conectar a internet. Esto nos daría algo así como 4 mil millones de posibles direcciones IP.

Se subdivide en  $X.Y.Z.W$  Donde  $X, Y, Z, W$  son un byte que numera del 0 al 255.

Dentro de las características de una IP tenemos diferentes clasificaciones de la A-E, Se definen por el primero octeto de bits:

Con esto sabemos que las clases:

*A* va de 0.0.0.0- 127.255.255.255

*B* va de 128.0.0.0- 191.255.255.255

*C* va de 192.0.0.0- 223.255.255.255

*D* va de 224.0.0.0- 239.255.255.255

*E* va de 240.0.0.0- 255.255.255.255

La clase E está reservada para Estudio y Desarrollo y la clase D para Multicast.

### 2.0.1. Problemas con IPv4

Ahora, recuerda que te dije que IP v4 acepta unos 4 mil millones de direcciones válidas, ahora el problema es que ahora mismo hay vivos más de 7 mil millones de personas (A principios del siglo XXI) cada una con seguramente más de un dispositivo que quieren

conectar a internet.

Por lo tanto tenemos que encontrar una forma de solucionar esto.

### 3. Software

- Librería `stdlib.h`
- Librería `stdio.h`
- Librería `stdbool.h`

### 4. Procedimiento

- Leemos la ip, buscamos darle formato con el mismo escaner de la entrada, checamos que cada uno de los cuatro valores esten entre 0 y 225.
- Buscamos la manera más eficiente de guardar esta información que en este caso es un entero sin signo, colocamos la información y pasamos a procesarla.
- Checamos la clase, esto se puede hacer con los bits más significativos del cuarteto de bytes. Podemos darnos cuenta de que lo importante es ver que bits estan en 1 y cuales no, asi que podemos checarlos fácilmente. Para la clase A, el primer bit debe de ser 0. Para la clase B el segundo bit debe de ser 0, y así sucesivamente.
- Despues de esto la marcara la podemos obtener por definición de cada clase y con ella realizamos un OR con la IP y la Mascara negada.
- Realizamos el calculo de la Red con un and binario.
- El rango se obtiene simplemente sumando a la Red +1 y restando al Broadcast -1.
- Sabemos que tipo es por una comparación sencilla, comparamos con la red y el broadcast, si ninguno entra entonces es un tipo host.
- Por ultimo creamos una funcion que nos muestre por pantalla la IP con su formato.

### 5. Resultados

Tenemos los siguientes resultados:

```
lalaandrea10@lalaandrea10:~/Documentos/Redes/Practica1$ ./a.out
IP Calculator v1.0
Please give me your IP
1.1.1.1

Class A
Mask
255.0.0.0

Broadcast:
1.255.255.255
Network
1.0.0.0
Range:
1.0.0.1
-
1.255.255.254

Its an IP Host
```

```
lalaandrea10@lalaandrea10:~/Documentos/Redes/Practica1$ ./a.out
IP Calculator v1.0
Please give me your IP
129.0.2.4

Class B
Mask
255.255.0.0

Broadcast:
129.0.255.255
Network
129.0.0.0
Range:
129.0.0.1
-
129.0.255.254

Its an IP Host
```

```
IP Calculator v1.0
Please give me your IP
192.4.6.7

Class C
Mask
255.255.255.0

Broadcast:
192.4.6.255
Network
192.4.6.0
Range:
192.4.6.1
-
192.4.6.254

Is a IP Host
```

```
lalaandrea10@lalaandrea10:~/Documentos/Redes/Practica1$ ./a.out
IP Calculator v1.0
Please give me your IP
224.5.6.7
Class D
Multicast
The range its from 224.0.0.0 to 239.255.255.255
it is reservated for multicast.
lalaandrea10@lalaandrea10:~/Documentos/Redes/Practica1$ ./a.out
IP Calculator v1.0
Please give me your IP
240.0.0.0
Class E
Reserch and development
This IP class is reservated for experimentation
only for R&D or study. The IP adress for this class
its from 240.0.0.0 to 255.255.255.254
lalaandrea10@lalaandrea10:~/Documentos/Redes/Practica1$
```

## 6. Discusión

Al iniciar la aplicación nos encontramos con la bienvenida y pide inmediatamente los datos, después realiza el cálculo y nos muestra la clase con su respectiva máscara de default, después nos muestra los demás aspectos de la IP, el broadcast y la red.

Realizar este tipo de cálculos manualmente llega a ser algo peligroso, el equivocarte con algún dígito puede hacer que tus redes queden mal y causen problemas, además de que cuando lo realizas, pierdes tiempo, hacer esto es realmente más sencillo, y acertado.

to explore the significance of your study's findings. qualify and explore the theoretical importance/significance of your results.

The discussion is also the place in a report where any qualifications or reservations you have about the research should be aired.

## 7. Conclusión

Aprender a realizar cálculos bit a bit fue bastante interesante, normalmente nos dicen ese tipo de cosas pero al no aplicarlas no vemos el gran potencial que pueden tener.

Por ejemplo usar un unsigned int para almacenar 32 bits de una IP me parece realmente interesante, o usar los corrimientos y los operadores binarios una manera sencilla, elegante y eficiente de realizar este tipo de cálculos.

Cuando lo implemente tuve varios errores, algunos más difíciles de corregir que otros, uno que me costó trabajo ver fue la jerarquía de los operadores binarios, me realizaba cosas extrañas.

La función que se me hizo difícil fue mostrarlo, pues lo guarde como un entero completo, si embargo realizar los cálculos fue muy sencillo, esta manera me gustó más que realizar 4 variables para una IP.

Uno de los errores de mi programa es cuando le mandas caracteres, no coloca nada dentro de mi IP entonces se queda con el 0.0.0.0 default.

Una última mejora a realizar es la comparación para definir la clase puedo realizarla de mejor manera sin necesidad de realizar un corrimiento y además la operación. Veremos esta mejora en la siguiente versión de esta práctica.

## 8. Código

```
1 | #include <stdlib.h>
2 | #include <stdio.h>
3 | #include <stdbool.h>
4 | typedef unsigned int uint;
5 | typedef unsigned char byte;
```

```

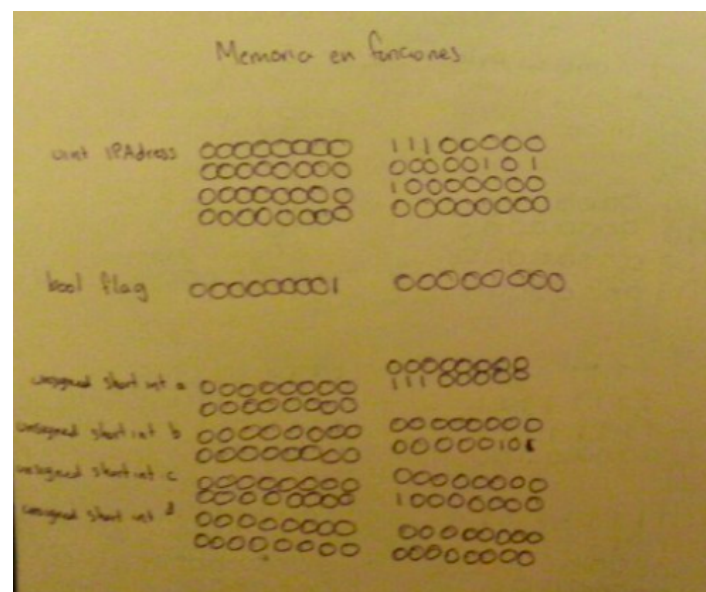
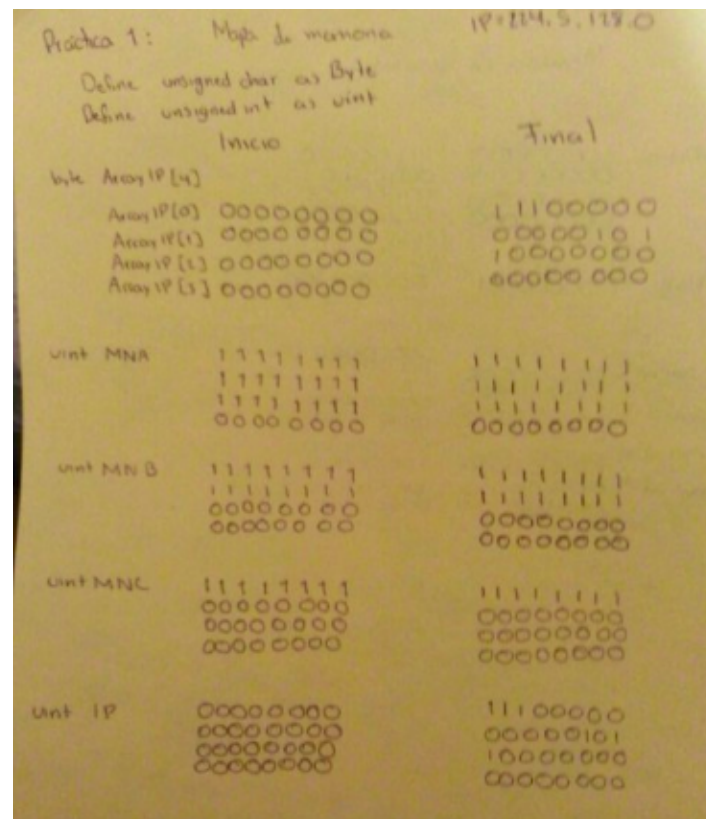
6
7     byte ArrayIP[4]={0,0,0,0};
8
9  *****Only for reserch uses*****
10 void binprintf(int IP) //=====PRINT AS BINARY
11 {
12     unsigned int aux=1<<((sizeof(int)<<3)-1); //Change the pointer
13     while(aux) { //While aux is true
14         printf("%d", (IP&aux ? 1 : 0)); //Print it
15         aux >>= 1; //Move it baby
16     }
17 }
18 *****Only for reserch uses*****
19 //=====HELLO DARKNESS MY OLD FRIEND=====
20 //scanf("%hu.%hu.%hu.%hu", &ArrayIP[0], &ArrayIP[1], &ArrayIP[2], &ArrayIP[3]); //What is
21 your IP?
22 printf("IP\n");
23 ToIPPrint(IP);
24 printf("\n");
25 *****Only for reserch uses*****
26 uint CreateIP(byte ArrayIP[4]) { //=====CREATE AN IP=====
27     uint IPAddress = 0; //Start by cleaning it all*****
28     IPAddress |= ArrayIP[0] << (32 - 8); //Let's put a info in 0-7
29     IPAddress |= ArrayIP[1] << (32 - 16); //Let's put b info in 8-15
30     IPAddress |= ArrayIP[2] << (32 - 24); //Let's put c info in 16-31
31     IPAddress |= ArrayIP[3]; //The info in d just fit
32
33     return IPAddress; //Go little butterfly
34 }
35
36 void Scan() { //Scan my elements
37     bool flag=true; //We will need this
38     unsigned short int a = 0 ,b=0,c=0,d=0;
39
40     printf("IP Calculator v1.0\nPlease give me your IP\n");
41
42     do{
43         scanf("%u.%u.%u.%u", &a, &b, &c, &d); //Scan my adress
44         if(a<=255 && b<=255 && c<=255 && d<=255) { //If youre pretty
45             ArrayIP[0]=a; //Push into my bytes
46             ArrayIP[1]=b; //Push into my bytes
47             ArrayIP[2]=c; //Push into my bytes
48             ArrayIP[3]=d; //Push into my bytes
49             flag=false; //Let me out
50         }
51         else printf("Wrong IP, The values are between 0-255 please try again.\n"); //Let me know im fool
52     }while(flag);
53
54 }
55 void ToIPPrint(uint IP) //=====Let's print it pretty
56 {
57     printf("%d.%d.%d.%d\n", ((IP >> 24) & 0xFF), ((IP >> 16) & 0xFF), ((IP >> 8) & 0xFF), (IP & 0xFF)); //Print it
58 }
59
60
61 int main(int argc, char **argv)
62 {
63
64     uint MNC= 0b11111111111111111111111100000000; //Initialize the mask Class A
65     uint MNB= 0b1111111111111111111111110000000000000000; //Initialize the mask Class B
66     uint MNA= 0b111111111111111111111111000000000000000000000000000; //Initialize the mask Class C
67
68     Scan(); //Scan my IP
69     uint IP=CreateIP(ArrayIP); //Give me my IP
70
71     //=====If youre a Class A=====
72     if(((IP>>31)&1)==0){
73         printf("\nClass A\n"); //Print your class
74         printf("Mask\n"); // and your mask
75         ToIPPrint(MNA);
76
77
78         printf("\nBroadcast: \n"); //Print your boradcast
79         ToIPPrint(IP|~(MNA));
80
81         printf("Network\n"); //Print you Network
82         ToIPPrint(IP&MNA);
83
84         printf("Range:\n"); //Print your Range
85         ToIPPrint((IP&MNA)+1);
86         printf("-\n");
87         ToIPPrint((IP|~(MNA))-1);
88
89

```

```
90     printf("\n");
91     if (IP==(IP&MNA)) printf("Its an IP Network\n");
92     else if (IP==(IP|^(MNA))) printf("Its an IP Broadcast\n");
93     else printf("Its an IP Host\n");
94
95 }
96 else if (((IP>>30) & 1) == 0){
97     printf("\nClass B\n");
98     printf("Mask\n");
99     ToIPPrint(MNB);
100
101     printf("\nBroadcast: \n");
102     ToIPPrint((IP|^(MNB)));
103
104     printf("Network\n");
105     ToIPPrint(IP&MNB);
106
107     printf("Range:\n");
108     ToIPPrint((IP&MNB)+1);
109     printf("\n\n");
110     ToIPPrint((IP|^(MNB))-1);
111
112
113     printf("\n");
114     if (IP==(IP&MNB)) printf("Its an IP Network\n");
115     else if (IP==(IP|^(MNB))) printf("Its an IP Broadcast\n");
116     else printf("Its an IP Host\n");
117
118 }
119 else if (((IP>>29) & 1) == 0){
120     printf("\nClass C\n");
121     printf("Mask\n");
122     ToIPPrint(MNC);
123
124
125     printf("\nBroadcast: \n");
126     ToIPPrint(IP|^(MNC));
127
128     printf("Network\n");
129     ToIPPrint(IP&MNC);
130
131     printf("Range:\n");
132     ToIPPrint((IP&MNC)+1);
133     printf("\n\n");
134     ToIPPrint((IP|^(MNC))-1);
135
136
137     printf("\n");
138     if (IP==(IP&MNC)) printf("Its an IP Network\n");
139     else if (IP==(IP|^(MNC))) printf("Its an IP Broadcast\n");
140     else printf("Is a IP Host\n");
141
142 }
143 else if (((IP>>28) & 1) == 0){
144     printf("Class D\n");
145     printf("Multicast\n");
146     printf("The range its from 224.0.0.0 to 239.255.255.255 \n it is reservated for multicast.\n");
147
148 }
149
150 else{
151     printf("Class E\n");
152     printf("Reserch and development\n");
153     printf("This IP class is reservated for experimentation\nonly for R&D or study. The IP\n");
154     printf("adress for this class\nits from 240.0.0.0 to 255.255.255.254\n");
155
156 }
157
158
159 }
```

## 9. Mapa de memoria





## Referencias

- [1] Axel Ernesto Moreno Cervantes *Redes de Computación*. ESCOM, 2018.
- [2] Nidia Cortez. *Redes de Computadoras* ESCOM, 2018.