

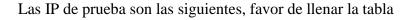
Nombre \_\_\_\_\_





Objetivo: el alumno programará una calculadora IP versión 1 en C. Haciendo uso de un block de notas y compilando en gcc

Dada una dirección IP (inicializada en hexadécimal) Output: Clase, Dirección IP de Red y dirección de Broadcast.





	Dirección IP	Dir en hexadecimal	Clase	Dir. IP Red	Broadcast
Z	<mark>10</mark> .250.1.1	0x0A.0xFA.0x01.0x0	А	10.0.0.0	10.255.255.255
		1			
Y	150.10 <mark>.15.0</mark>	0X96 0X0A,0X0F	В	<mark>150.10</mark> . <mark>0.0</mark>	150.10.255.255
		0X00			
X	192.14.2. <mark>0</mark>	c0.e.2.0	С	192.14.2. <mark>0</mark>	192.14.2.255
W	148.17.9.1	94.11.9.1	В	148.17.0.0	148.17.255.255
V	193.0.0. <mark>255</mark>	c1.0.0.ff	С	193.0.0.0	193.0.0.255
U	220.200.23.1	dc.c8.17.1	С	220.200.23.0	220.200.23.255
T	177.100.18.4	b1.64.12.4	В	177.100.0.0	177.100.255.255
S	95. <mark>0.21.0</mark>	5f.0.15.0	А	95.0.0.0	95.255.255.255
R	111.111.111.111	6f.6f.6f.6f	А	111.0.0.0	111.255.255.255
Q	219.26.51.0	db.1a.33.0	С	219.26.51.0	219.26.51.255
P	<mark>167.0</mark> .0.0	a7.0.0.0	В	167.0 <mark>.0.0</mark>	<mark>167.0</mark> .255.255
0	240.1.1.1	f0.1.1.1	E	-	_
N	<mark>10</mark> . <mark>1.1.1</mark>	a.1.1.1	А	10.0.0.0	10.255.255.255
M	225.1.1.1	e1.1.1.1	D	_	_

#### Instrucciones.

- -Abrir un block de notas y escribir el código correspondiente en C.
- -La dirección IP se inicializará en hexadecimal en un arreglo de caracteres sin signo
- La mascara de red se inicializará en décimal en un arreglo de caracteres sin signo
- -Compilar en consola haciendo uso de gcc (en caso de no tenerlo, instalarlo).
- -Una vez terminado deberán probar con las 14 direcciones IP de la tabla. Tomar captura de pantalla de la salida para cada IP y pegarlas al final de este documento.





Nombre \_\_\_\_\_





Se entrega este documento, incluir:

#### Todo tu Código

#include <stdlib.h>

#include <stdio.h>

```
unsigned char* Clase_Y_Mascara(unsigned char *IP0,unsigned char* limInf);
void Rango(unsigned char* IP,unsigned char* MASK);
void Tipo(unsigned char* IP,unsigned char *MASK);
int main(){
   unsigned char IP[4],limInf;
   unsigned char* MASK;
   printf("Ingrese La IP: ");
   scanf("%x.%x.%x.%x",&IP[0],&IP[1],&IP[2],&IP[3]);
   printf("IPv4: %.2x.%.2x.%.2x.%.2x\n",IP[0],IP[1],IP[2],IP[3]);
   MASK = Clase_Y_Mascara(&IP[0],&IImInf);
   if (MASK[0]){printf("Mascara de subred: %.2x.%.2x.%.2x.%.2x\n",MASK[0],MASK[1],MASK[2],MASK[3]);
}else printf("No tiene mascara de subred default!!\n");
Rango(IP,MASK);
```

### return 0;

```
unsigned char* Clase_Y_Mascara(unsigned char *IPO,unsigned char* limInf){
unsigned char* Mask = (unsigned char*)calloc(4,sizeof(unsigned char));
if (*IPO&128){
if(*IPO&32){
if(*IPO&32){
if(*IPO&32){
printf("Clase: E\n");Mask[0] = 0;Mask[1] = 0;Mask[2] = 0;Mask[3] = 0;*limInf = 240;
}else {printf("Clase: D\n"); Mask[0] = 0;Mask[1] = 0;Mask[2] = 0;Mask[3] = 0;*limInf = 224;}
}else {printf("Clase: C\n"); Mask[0] = 255;Mask[1] = 255;Mask[2] = 255;Mask[3] = 0;*limInf = 192;}
}else {printf("Clase: B\n"); Mask[0] = 255;Mask[1] = 255;Mask[2] = 0;Mask[3] = 0;*limInf = 128;}
}else {printf("Clase: A\n"); Mask[0] = 255;Mask[1] = 0;Mask[2] = 0;Mask[3] = 0;*limInf = 0;}
return Mask;
}
```

void Rango(unsigned char\* IP,unsigned char\* MASK){
unsigned char\* Auxiliar = (unsigned char\*)calloc(4,1);



Nombre \_\_\_\_\_





}else{printf("SIN MASCARA!\n");}

Deberan crear y mostrar **el mapa de memoria** utilizado en su programa (considerar registros de 8 bits)

Ejemplo de mapa de memoria

,	Transfer in the								
	Unsigned char IP[0]	0	0	1	0	0	0	1	0
		0							
1			0	0	0	0	0	0	0
		0							
	2		0	0	0	0	0	0	0
		0							
	3		0	0	0	0	0	0	0
	Unsigned char MR[0]	1	1	1	1	1	1	1	1
1		0	0	0	0	0	0	0	0
2		0	0	0	0	0	0	0	0
3		0	0	0	0	0	0	0	0

Esto para: 10.1.1.1 MR: 255.0.0.0

Finalmente incluir tus Conclusiones individuales

\* sobre el uso de

If ()

Else como falso y verdadero

Dra. Nidia A. Cortez Duarte



Nombre \_\_\_\_\_





#### Comparar el costo computacional entre.

#### Contra

¿Habías usado los operadores binarios? ¿Qué opinas del uso de operadores binarios?





Nombre \_\_\_\_\_





#### Conclusiones

#### Barros Luis enrique

Al usar if-else como un verdadero falso logramos ser concretos en usar condiciones especificas para las deciciones tomadas por el programa, siendo precisas las características calculadas con las IP's

Ambas expresiones son bastante simples y no presentan una complejidad computacional significativa. El costo de ejecución de ambas expresiones puede depender mas del lenguaje en el que se programe y/o compile.

Habia tenido previo uso de los operadores binarios en el uso de controladores en vocacional.

Me parece importante utilizarlo, pues gasta menos memoria.

#### Bautista Alfredo

Tener if-else al trabajar en un bajo nivel computacional puede ser de gran utilidad, logra llegar a resultados precisos siempre y cuando el programador lo prepare para ello, sin llegar a usar memoria de mas.

El uso de (&) podria tener un menor costo computacional, a nivel bits son mas rapidas que comparaciones estandar, sin embargo, seria imperceptible la diferencia.

No habia tenido un acercamiento con este nivel de programacion, me parecio de gran utilidad y lo contemplo para futuras aplicaciones por su rapidez y poco uso de memoria.

Criterio	Valor	Tu evaluación
El programa se escribe en un block de notas y se compila con gcc	1	1
Se incluye todo el código y se hace uso de operadores binarios	1	1
El mapa de memoria refleja todas las variables utilizadas en el programa y han sido seleccionadas	1	1
de forma consciente.		
Las imágenes son claras	1	1
Se incluyen las conclusiones solicitadas	1	1
TOTAL	5	5

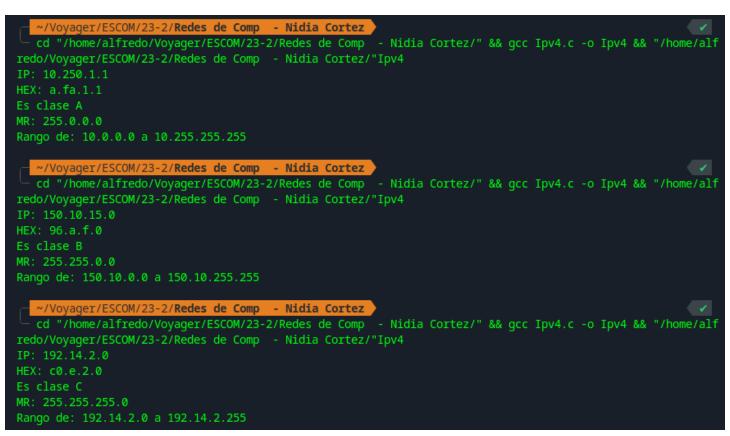




Nombre \_\_\_\_\_







#### rager/ESCOM/23-2/**Redes de Comp - Nidia Cortez** cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4 IP: 148.17.9.1 HEX: 94.11.9.1 Es clase B MR: 255.255.0.0 Rango de: 148.17.0.0 a 148.17.255.255 ger/ESCOM/23-2/**Redes de Comp - Nidia Cortez** cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4 IP: 193.0.0.255 HEX: c1.0.0.ff Es clase C MR: 255.255.255.0 Rango de: 193.0.0.0 a 193.0.0.255



S



Nombre \_\_\_\_\_





~/Voyager/ESCOM/23-2/**Redes de Comp - Nidia Cortez** 

INT X

cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4

IP: 220.200.23.1 HEX: dc.c8.17.1 Es clase C

MR: 255.255.255.0

Rango de: 220.200.23.0 a 220.200.23.255

~/Voyager/ESCOM/23-2/**Redes de Comp - Nidia Cortez** 

- cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf

redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4

IP: 177.100.18.4 HEX: b1.64.12.4 Es clase B MR: 255.255.0.0

Rango de: 177.100.0.0 a 177.100.255.255

~/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez

cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf

redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4

IP: 95.0.21.0 HEX: 5f.0.15.0 Es clase A MR: 255.0.0.0

Rango de: 95.0.0.0 a 95.255.255.255



Dra. Nidia A. Cortez Duarte



Nombre \_\_\_\_\_





ger/ESCOM/23-2/Redes de Comp - Nidia Cortez cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4 IP: 111.111.111.111 HEX: 6f.6f.6f.6f MR: 255.0.0.0 Rango de: 111.0.0.0 a 111.255.255.255 ager/ESCOM/23-2/**Redes de Comp - Nidia Cortez** cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4 IP: 219.26.51.0 HEX: db.1a.33.0 Es clase C MR: 255.255.255.0 Rango de: 219.26.51.0 a 219.26.51.255 ager/ESCOM/23-2/**Redes de Comp - Nidia Cortez** cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4 IP: 167.0.0.0 HEX: a7.0.0.0 Es clase B MR: 255.255.0.0 Rango de: 167.0.0.0 a 167.0.255.255 ger/ESCOM/23-2/**Redes de Comp - Nidia Cortez** cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alf redo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4 IP: 240.1.1.1 HEX: f0.1.1.1 Es clase E Rango de: 240.0.0.0 a 240.255.255.255

"/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez

cd "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/" && gcc Ipv4.c -o Ipv4 && "/home/alfredo/Voyager/ESCOM/23-2/Redes de Comp - Nidia Cortez/"Ipv4
IP: 10.1.1.1
HEX: a.1.1.1

HEX: a.1.1.1 Es clase A MR: 255.0.0.0

Rango de: 10.0.0.0 a 10.255.255.255



Dra. Nidia A. Cortez Duarte