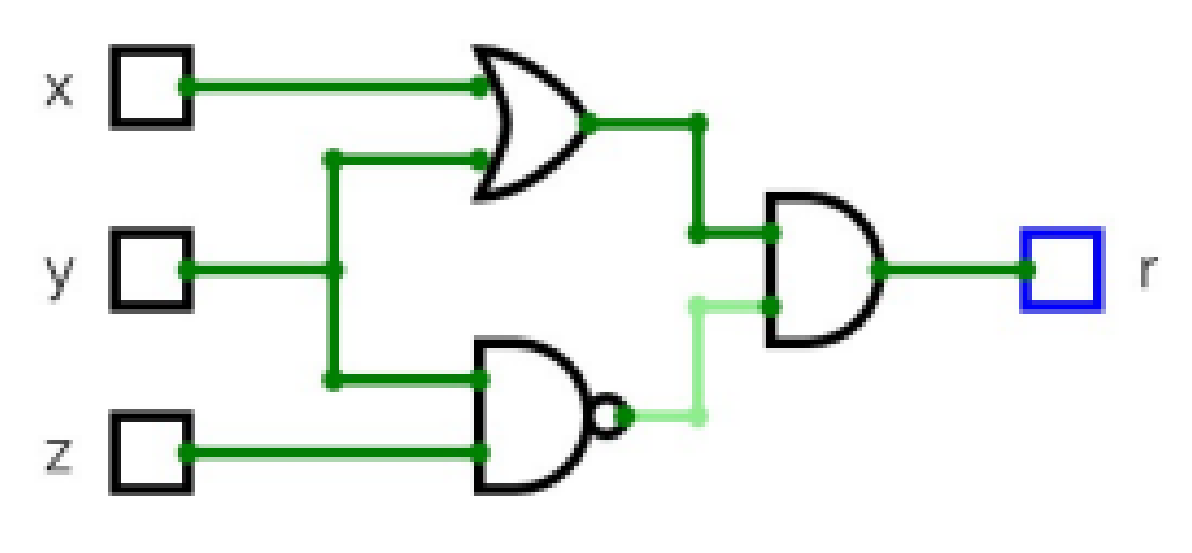
Quiz de Laboratorio #1

Juan Bonte Hernández (22-10051)

Sea D el circuito digital a continuación:



1. Construya la tabla de verdad de D:

x	y	z	NOR(x,y)	AND(y,z)	r
1	1	1	0	1	**1**
1	1	0	0	0	**0**
1	0	1	0	0	**0**
1	0	0	0	0	**0**
0	1	1	0	1	**1**
0	1	0	0	0	**0**
0	0	1	1	0	**1**
0	0	0	1	0	**1**

2. Implemente una función isomorfa al circuito D en Zilly:

```
Z ⇒ Z not := fn(Z p) → if(p, 0, 1);
Z ⇒ Z ⇒ Z and := fn(Z p) → fn(Z q) → if(p, if(q, 1, 0), 0);
Z ⇒ Z ⇒ Z or := fn(Z p) → fn(Z q) → if(p, 1, if(q, 1, 0));
Z ⇒ Z ⇒ Z nor := fn(Z a) → fn(Z b) → not(or(a)(b));
Z ⇒ Z ⇒ Z ⇒ Z D := fn(Z x) → fn(Z y) → fn(Z z) → or(nor(x)(y))(and(y)(z));
D(1)(1)(0)
D(1)(0)(1)
D(1)(0)(0)
D(0)(1)(1)
D(0)(1)(0)
D(0)(0)(1)
D(0)(0)(0)
```

Cuya salida esperada sería

```
Q: Z ⇒ Z ⇒ Z and := fn(Z p) → fn(Z q) → if(p, if(q, 1, 0), 0);
R: ACK: Z ⇒ Z ⇒ Z and := λ(Z p) → λ(Z q) → if( p, if( q, 1, 0), 0);
Q: Z ⇒ Z ⇒ Z or := fn(Z p) → fn(Z q) → if(p, 1, if(q, 1, 0));
R: ACK: Z ⇒ Z ⇒ Z or := λ(Z p) → λ(Z q) → if( p, 1, if( q, 1, 0));
Q: Z ⇒ Z ⇒ Z nor := fn(Z a) → fn(Z b) → not(or(a)(b));
R: ACK: Z ⇒ Z ⇒ Z nor := λ(Z a) → λ(Z b) → not(or(a)(b));
Q: Z ⇒ Z ⇒ Z ⇒ Z D := fn(Z x) → fn(Z y) → fn(Z z) → or(nor(x)(y))(and(y)(z));
R: ACK: Z ⇒ Z ⇒ Z ⇒ Z D := λ(Z x) → λ(Z y) → λ(Z z) → or(nor(x)(y))(and(y)(z));
Q: D(1)(1)(0)
R: OK: D(1)(1)(0) ⟹ 0
Q: D(1)(0)(1)
R: OK: D(1)(0)(1) ⟹ 0
Q: D(1)(0)(0)
R: OK: D(1)(0)(0) ⟹ 0
Q: D(0)(1)(1)
R: OK: D(0)(1)(1) ⟹ 1
Q: D(0)(1)(0)
R: OK: D(0)(1)(0) ⟹ 0
Q: D(0)(0)(1)
R: OK: D(0)(0)(1) ⟹ 1
Q: D(0)(0)(0)
R: OK: D(0)(0)(0) ⟹ 1
```

3. Implemente una función isomorfa al circuito D en C++

```
#include <iostream>
using namespace std;

int notZ(int p){ return p ? 0 : 1; }
int andZ(int p,int q){ return (p && q) ? 1 : 0; }
int orZ(int p,int q){ return (p || q) ? 1 : 0; }
int norZ(int a,int b){ return notZ(orZ(a,b)); }

int D(int x,int y,int z){ return orZ(norZ(x,y), andZ(y,z)); }

int main(){
   cout << "x y z | NOR | AND | r\n";
   cout << "----------------------\n";
   for(int x=1;x>=0;--x)
     for(int y=1;y>=0;--y)
       for(int z=1;z>=0;--z){
         int n = norZ(x,y), a = andZ(y,z), r = D(x,y,z);
         cout << x<<" "<<y<<" "<<z<<" |  "<<n<<"  |  "<<a<<"  | "<<r<<"\n";
       }
}
```

Cuya salida esperada es:

```
x y z | NOR | AND | r
----------------------
1 1 1 |  0  |  1  | 1
1 1 0 |  0  |  0  | 0
1 0 1 |  0  |  0  | 0
1 0 0 |  0  |  0  | 0
0 1 1 |  0  |  1  | 1
0 1 0 |  0  |  0  | 0
0 0 1 |  1  |  0  | 1
0 0 0 |  1  |  0  | 1
```

> Todos los archivos con el código están subidos en el repositorio:
> https://github.com/juanbonteh/CI2125