How they interact:

Bob learns if he can receive blood from Alice

Alice		Bob
		$(F, e, d) \leftarrow Gb(1^k, f, T)$
	$\xleftarrow{\quad F \quad}$	
		$Y \leftarrow En(e_y, y)$
	$\xleftarrow{\quad Y \quad}$	
$\xrightarrow{x_i}$	$\boxed{OT}$	$\xleftarrow{K_0^i, K_1^i}$ Run 3 times
$\xleftarrow{K_{x_i}^i}$		
$X = [K_{x_1}^1, K_{x_2}^2, K_{x_3}^3]$	Note this gives A $X \leftarrow En(e_x, x)$	
$Z \leftarrow Ev(F, (X \Vert Y))$		
	$\xrightarrow{\quad Z \quad}$	
		output $z \leftarrow De(d, Z)$

we note that $e = (e_{x_A}, e_{y_A}, e_{x_B}, e_{y_B}, e_{x_R}, e_{y_R})$ where

$e_x = (e_{x_A}, e_{x_B}, e_{x_R})$ and

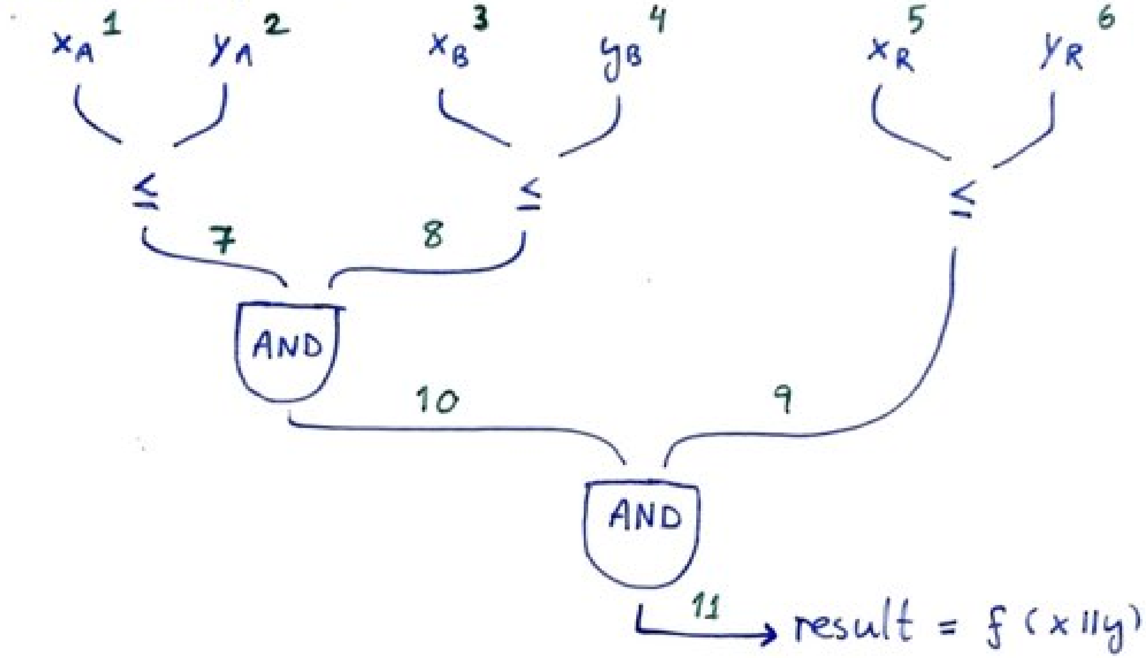
$e_y = (e_{y_A}, e_{y_B}, e_{y_R})$.

Circuit $f: \{0,1\}^6 \to \{0,1\}$, $T = 11$ wires

input on form $f(x \| y)$, see if y can receive blood from x.

$$f((x_A, x_B, x_R), (y_A, y_B, y_R)) = (((x_A \leq y_A) \cdot (x_B \leq y_B)) \cdot (x_R \leq y_R))$$

Visualized as follows:



Numbering of wires:

w_i for $i \in [1, .., 6]$ input wires

w_i for $i \in [7, 8, 9, 10]$ internal wires

w_T for $T = 11$ output wire.

Notation: $w_i = \text{Eval}(w_{L(i)}, w_{R(i)})$, where $L, R: [7, 11] \to [1, .., 11]$:

$w_7 = w_1 \leq w_2$, $w_8 = w_3 \leq w_4$, $w_9 = w_5 \leq w_6$,

$w_{10} = w_7 \wedge w_8$, $w_{11} = w_9 \wedge w_{10}$

Eval is either "$\leq$" or "$\wedge$". Note $L(i) \leq R(i) < i$.

$$L: \begin{array}{l} 7 \to 1 \\ 8 \to 3 \\ 9 \to 5 \\ 10 \to 7 \\ 11 \to 9 \end{array} \qquad R: \begin{array}{l} 7 \to 2 \\ 8 \to 4 \\ 9 \to 6 \\ 10 \to 8 \\ 11 \to 10 \end{array}$$

We need to have a PRF

$$G : \{0,1\}^k \times \{0,1\}^k \times [1, \dots, 11] \to \{0,1\}^{2 \cdot 128},$$

use SHA-256.

"set length of wire labels to 128 bits $\to k = 128$?"

Circuit generation : Want output (F, e, d)

$Gb(f, 11)$:

For i in range$(1, \overset{11+1}{12})$: // $i = 1, \dots, 11$

choose $(K_0^i, K_1^i) \xleftarrow{\$} \{0,1\}^k \times \{0,1\}^k$

All_key_values.append(K_0^i, K_1^i) // store <u>all</u> keys somewhere

if $i < 7$: // if $i \in \{1,2,3,4,5,6\}$

e.append$((K_0^i, K_1^i))$ // store keys for input wires in some way
de to access

if $i = 11$:

$d := (Z_0, Z_1) = (K_0^{11}, K_1^{11})$

Note that $e_x = \{K_0^i, K_1^i\}_{i \in \{1,3,5\}}$ will be used for Alice, and

$e_y = \{K_0^i, K_1^i\}_{i \in \{2,4,6\}}$ will be used for Bob

when encoding their input.

$\longrightarrow$

For i in range$(7, \overset{= 11+1}{12})$: // define $C_0^i, \dots, C_3^i$ for F

$$C_{00}^{i} = G(K_0^{L(i)}, K_0^{R(i)}, i) \oplus (K^{i}_{Eval(0,0)}, 0^{k})$$

$$C'_{01} = G(K_0^{L(i)}, K_1^{R(i)}, i) \oplus (K^{i}_{Eval(0,1)}, 0^{k})$$

$$C'_{10} = G(K_1^{L(i)}, K_0^{R(i)}, i) \oplus (K^{i}_{Eval(1,0)}, 0^{k})$$

$$C'_{11} = G(K_1^{L(i)}, K_1^{R(i)}, i) \oplus (K^{i}_{Eval(1,1)}, 0^{k})$$

choose random permutation $\pi : \{0,1,2,3\} \to \{0,1\} \times \{0,1\}$

$$(C_0^i, C_1^i, C_2^i, C_3^i) = (C'_{\pi(0)}, C'_{\pi(1)}, C'_{\pi(2)}, C'_{\pi(3)})$$

F. append$((C_0^i, C_1^i, C_2^i, C_3^i))$ L' 1

Encoding:

$En(e, x^{\in\{0,1\}^3})$:

Parse $e = [[K_0^1, K_1^1], \dots, [K_0^6, K_1^6]]$

For Alice:

Return $X = [K_{x_1}^1, K_{x_2}^3, K_{x_3}^5]$

For Bob:

Return $Y = [K_{x_1}, K_{x_2}^4, K_{x_3}^6]$

Evaluate:

$Ev(F, X, Y)$

Parse $X = [K^1, K^3, K^5]$, $Y = [K^2, K^4, K^6]$

K_values = $[K^1, K^2, K^3, K^4, K^5, K^6]$

For i in range(7, 11):

Access $(C_0^i, C_1^i, C_2^i, C_3^i)$

For j in range(4):

$(K'_j, \tau_j) = G(K^{L(i)}, K^{R(i)}, i) \oplus C_j^i$

save all values, check for correct result outside of loop since uniqueness

If unique j st $\tau_j = 0^k$:

$K^i = K'_j$, K_values.append(K^i)

Else: Abort, return $\perp$

Output $Z' = K^{11}$

Decoding:

$De(d, Z)$:

Parse $d = (Z_0, Z_1)$

If $Z = Z_0$:

Return 0

If $Z = Z_1$:

Return 1

else:

Return $\perp$