Lecture 21
Deterministic Communication Complexity

Benjamin Sach
Communication Complexity
Communication Complexity

$x$

$y$
Communication Complexity

this is Alice

\(x\)

this is Bob

\(y\)
Communication Complexity

this is Alice

this is Bob

Alice has a bit string $x \in X$
Communication Complexity

this is Alice

this is Bob

Bob has a bit string $y \in Y$

Alice has a bit string $x \in X$
Communication Complexity

this is Alice

this is Bob

Bob has a bit string $y \in Y$

Alice has a bit string $x \in X$

- Together they want to compute $z = f(x, y)$
Communication Complexity

this is Alice

this is Bob

X

Alice has a bit string $x \in X$

Y

Bob has a bit string $y \in Y$

- Together they want to compute $z = f(x, y)$ (where $f : X \times Y \rightarrow Z$)
Communication Complexity

this is Alice

this is Bob

Alice has a bit string \( x \in X \)

Bob has a bit string \( y \in Y \)

- Together they want to compute \( z = f(x, y) \)
  (where \( f : X \times Y \to Z \))

- Alice and Bob communicate by following a prearranged protocol
Together they want to compute $z = f(x, y)$

(where $f : X \times Y \rightarrow Z$)

Alice and Bob communicate by following a prearranged protocol
• Together they want to compute \( z = f(x, y) \) (where \( f : X \times Y \rightarrow Z \))

• Alice and Bob communicate by following a prearranged protocol
Together they want to compute $z = f(x, y)$ (where $f : X \times Y \to Z$)

Alice and Bob communicate by following a prearranged protocol
• Together they want to compute $z = f(x, y)$

(\text{where } f : X \times Y \rightarrow Z)$$

• Alice and Bob communicate by following a prearranged protocol
• Together they want to compute $z = f(x, y)$

(\text{where } f : X \times Y \to Z)

• Alice and Bob communicate by following a prearranged protocol
• Together they want to compute $z = f(x, y)$

(where $f : X \times Y \rightarrow Z$)

• Alice and Bob communicate by following a prearranged protocol
Together they want to compute $z = f(x, y)$

(where $f : X \times Y \rightarrow Z$)

Alice and Bob communicate by following a prearranged protocol

The protocol determines who communicates next and what they send
Together they want to compute \( z = f(x, y) \) (where \( f : X \times Y \to Z \))

- Alice and Bob communicate by following a prearranged protocol
- The protocol determines who communicates next and what they send
- The cost of a protocol on a particular \((x, y)\) is the number of bits sent
• Together they want to compute $z = f(x, y)$ (where $f : X \times Y \to Z$)

• Alice and Bob communicate by following a prearranged protocol

• The protocol determines who communicates next and what they send

• The cost of a protocol on a particular $(x, y)$ is the number of bits sent

• The (overall) cost of a protocol is the maximum cost on any $(x, y)$
The communication complexity of $f$, $D(f)$ is the minimum cost of any protocol $\mathcal{P}$, which correctly computes $f$. 
The communication complexity of $f$, $D(f)$ is the minimum cost of any protocol $\mathcal{P}$, which correctly computes $f$.

It's the minimum number of bits of communication Alice and Bob need to communicate to compute $f(x, y)$ (for any $(x, y)$).
The communication complexity of $f$, $D(f)$ is the minimum cost of any protocol $\mathcal{P}$, which correctly computes $f$.

It's the minimum number of bits of communication Alice and Bob need to communicate to compute $f(x, y)$ (for any $(x, y)$).

We will be interested in proving upper and lower bounds on $D(f)$ for a given $f$. 

• The communication complexity of $f$, $D(f)$ is the minimum cost of any protocol $\mathcal{P}$, which correctly computes $f$.

• It's the minimum number of bits of communication Alice and Bob need to communicate to compute $f(x, y)$ (for any $(x, y)$).

• We will be interested in proving upper and lower bounds on $D(f)$ for a given $f$. 

Diagram:

- A graph with nodes labeled $x$ and $y$.
- Edges between nodes labeled with 0 and 1.
Communication Complexity

What is all this good for?
Communication Complexity

What is all this good for?

- There are lots of immediate applications...
Communication Complexity

$\chi$  

What is all this good for?

- There are lots of immediate applications. . .
  - optimisation of computer networks
Communication Complexity

What is all this good for?

- There are lots of immediate applications. . .
  - optimisation of computer networks
  - communication between multiple cores on a CPU
Communication Complexity

What is all this good for?

- There are lots of immediate applications. . .
  - optimisation of computer networks
  - communication between multiple cores on a CPU
  - and basically anything involving the internet :)
What is all this good for?

• There are lots of immediate applications. . .
  ○ optimisation of computer networks
  ○ communication between multiple cores on a CPU
  ○ and basically anything involving the internet :)

• There are many more less immediate applications. . .
What is all this good for?

- There are lots of immediate applications...  
  - optimisation of computer networks
  - communication between multiple cores on a CPU
  - and basically anything involving the internet :)

- There are many more less immediate applications...

  *Particularly as a tool for algorithm and data structure lower bounds*
Protocol Trees

We can define a protocol formally as a binary tree…

\[
\begin{align*}
  a_1(00) &= 0 \\
  a_1(01) &= 0 \\
  a_1(10) &= 1 \\
  a_1(11) &= 1 \\
  b_2(00) &= 0 \\
  b_2(01) &= 0 \\
  b_2(10) &= 0 \\
  b_2(11) &= 1 \\
  b_3(00) &= 1 \\
  b_3(01) &= 0 \\
  b_3(10) &= 0 \\
  b_3(11) &= 0 \\
  a_4(00) &= 0 \\
  a_4(01) &= 0 \\
  a_4(10) &= 0 \\
  a_4(11) &= 1
\end{align*}
\]

<table>
<thead>
<tr>
<th>f</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Protocol Trees

We can define a protocol formally as a binary tree…

But first we’ll look at an example of a problem and a protocol
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \]

\[ y = y_1 y_2 y_3 \ldots y_n \]
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \quad \text{and} \quad y = y_1 y_2 y_3 \ldots y_n \]

• Alice and Bob want to compute
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \quad \text{and} \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]
A first example: OR

\[ x = x_1x_2x_3 \ldots x_n \quad \text{and} \quad y = y_1y_2y_3 \ldots y_n \]

- Alice and Bob want to compute
  \[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]

Protocol 1:
  - Alice sends \( x \) to Bob
  - Bob computes \( z = f(x, y) \)
  - Bob sends \( z \) to Alice
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \]
\[ y = y_1 y_2 y_3 \ldots y_n \]

Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]

Protocol 1:

- Alice sends \( x \) to Bob
- Bob computes \( z = f(x, y) \)
- Bob sends \( z \) to Alice
A first example: OR

\[ x = x_1x_2x_3 \ldots x_n \]

\[ y = y_1y_2y_3 \ldots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]

**Protocol 1:**

- Alice sends \( x \) to Bob
- Bob computes \( z = f(x, y) \)
- Bob sends \( z \) to Alice
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \vee x_2 \vee \ldots \vee x_n \vee y_1 \vee y_2 \vee \ldots \vee y_n \]

**Protocol 1:**
- Alice sends \( x \) to Bob
- Bob computes \( z = f(x, y) \)
- Bob sends \( z \) to Alice

- This protocol takes \( n + 1 \) bits (and it’s correct)
A first example: OR

\[ x = x_1 x_2 x_3 \cdots x_n \]
\[ y = y_1 y_2 y_3 \cdots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \cdots \lor x_n \lor y_1 \lor y_2 \lor \cdots \lor y_n \]

**Protocol 1:**
- Alice sends \( x \) to Bob
- Bob computes \( z = f(x, y) \)
- Bob sends \( z \) to Alice

- This protocol takes \( n + 1 \) bits (and it’s correct)
- For any \( f : X \times Y \rightarrow Z \) we have that \( D(f) \leq \log |X| + \log |Z| \)
A first example: **OR**

\[
x = x_1 x_2 x_3 \ldots x_n
\]

\[
y = y_1 y_2 y_3 \ldots y_n
\]

- Alice and Bob want to compute

\[
f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n
\]
A first example: OR

\[ x = x_1x_2x_3 \ldots x_n \quad \text{and} \quad y = y_1y_2y_3 \ldots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]

**Protocol 2:**

- Alice computes \( g(x) = x_1 \lor x_2 \lor \ldots \lor x_n \)
- Alice sends \( g(x) \) to Bob
- Bob computes \( z = f(x, y) = g(x) \lor y_1 \lor y_2 \lor \ldots \lor y_n \)
- Bob sends \( z \) to Alice
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \]
\[ y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]

**Protocol 2:**

- Alice computes \( g(x) = x_1 \lor x_2 \lor \ldots \lor x_n \)
- Alice sends \( g(x) \) to Bob
- Bob computes \( z = f(x, y) = g(x) \lor y_1 \lor y_2 \lor \ldots \lor y_n \)
- Bob sends \( z \) to Alice
A first example: OR

\[ x = x_1 x_2 x_3 \ldots x_n \]

\[ y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute

\[ f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n \]

Protocol 2:

- Alice computes \( g(x) = x_1 \lor x_2 \lor \ldots \lor x_n \)
- Alice sends \( g(x) \) to Bob
- Bob computes \( z = f(x, y) = g(x) \lor y_1 \lor y_2 \lor \ldots \lor y_n \)
- Bob sends \( z \) to Alice
A first example: $\text{OR}$

$x = x_1x_2x_3 \ldots x_n$

$y = y_1y_2y_3 \ldots yn$

Alice and Bob want to compute

$$f(x, y) = x_1 \lor x_2 \lor \ldots \lor x_n \lor y_1 \lor y_2 \lor \ldots \lor y_n$$

Protocol 2:

- Alice computes $g(x) = x_1 \lor x_2 \lor \ldots \lor x_n$
- Alice sends $g(x)$ to Bob
- Bob computes $z = f(x, y) = g(x) \lor y_1 \lor y_2 \lor \ldots \lor y_n$
- Bob sends $z$ to Alice

- This protocol takes only 2 bits (and it’s correct)
We now give a formal definition of a protocol $P$ as a binary tree...

\[
\begin{align*}
    a_1(00) &= 0 \\
    a_1(01) &= 0 \\
    a_1(10) &= 1 \\
    a_1(11) &= 1 \\
    b_2(00) &= 0 \\
    b_2(01) &= 0 \\
    b_2(10) &= 0 \\
    b_2(11) &= 1 \\
    b_3(00) &= 1 \\
    b_3(01) &= 0 \\
    b_3(10) &= 0 \\
    b_3(11) &= 0 \\
    a_4(00) &= 0 \\
    a_4(01) &= 0 \\
    a_4(10) &= 0 \\
    a_4(11) &= 1
\end{align*}
\]
We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...
• We now give a formal definition of a protocol $P$ as a binary tree...

This protocol tree computes this $f$

\[
\begin{array}{c|cccc}
 f & 00 & 01 & 10 & 11 \\
\hline
 00 & 0 & 0 & 0 & 1 \\
 01 & 0 & 0 & 0 & 1 \\
 10 & 0 & 0 & 0 & 0 \\
 11 & 0 & 1 & 1 & 1 \\
\end{array}
\]
Protocol Trees

- We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

\[
\begin{align*}
  a_1(00) &= 0 \\
  a_1(01) &= 0 \\
  a_1(10) &= 1 \\
  a_1(11) &= 1 \\
  b_2(00) &= 0 \\
  b_2(01) &= 0 \\
  b_2(10) &= 0 \\
  b_2(11) &= 1 \\
  b_3(00) &= 1 \\
  b_3(01) &= 0 \\
  b_3(10) &= 0 \\
  b_3(11) &= 0 \\
  a_4(00) &= 0 \\
  a_4(01) &= 0 \\
  a_4(10) &= 0 \\
  a_4(11) &= 1
\end{align*}
\]

the protocol starts at the root

\[
\begin{array}{c|cccc}
  f & 00 & 01 & 10 & 11 \\
  \hline
  00 & 0 & 0 & 0 & 1 \\
  01 & 0 & 0 & 0 & 1 \\
  10 & 0 & 0 & 0 & 0 \\
  11 & 0 & 1 & 1 & 1 \\
\end{array}
\]

Bob
We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree:

- $a_1(00) = 0$
- $a_1(01) = 0$
- $a_1(10) = 1$
- $a_1(11) = 1$

The function $a_1 : X \rightarrow \{0, 1\}$ tells Alice what to send.

- $b_2(00) = 0$
- $b_2(01) = 0$
- $b_2(10) = 0$
- $b_2(11) = 1$

- $b_3(00) = 1$
- $b_3(01) = 0$
- $b_3(10) = 0$
- $b_3(11) = 0$

- $a_4(00) = 0$
- $a_4(01) = 0$
- $a_4(10) = 0$
- $a_4(11) = 1$

The table below shows the values of $f$ and $y$ for Bob:

<table>
<thead>
<tr>
<th>$f$</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The values of $x$ are shown in the diagram.
Protocol Trees

• We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

The bit sent dictates the path taken

<table>
<thead>
<tr>
<th></th>
<th>$f$</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Protocol Trees

• We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

\[
\begin{align*}
  a_1(00) &= 0 \\
  a_1(01) &= 0 \\
  a_1(10) &= 1 \\
  a_1(11) &= 1 \\
  b_3 : Y &\rightarrow \{0, 1\} \text{ tells Bob what to send}
\end{align*}
\]

\[
\begin{array}{c|ccc|c}
 f & 00 & 01 & 10 & 11 \\
\hline
 00 & 0 & 0 & 0 & 1 \\
 01 & 0 & 0 & 0 & 1 \\
 10 & 0 & 0 & 0 & 0 \\
 11 & 0 & 1 & 1 & 1 \\
\end{array}
\]
We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

Each node $v$ has a function $a_v$ or $b_v$ which dictates who sends a bit (and what to send)
Protocol Trees

- We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

Both Alice and Bob know the entire tree
We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

The protocol ends when a leaf node is reached (which gives $f(x, y)$)
Protocol Trees

• We now give a formal definition of a protocol $P$ as a binary tree...

\[
\begin{align*}
  a_1(00) &= 0 \\
  a_1(01) &= 0 \\
  a_1(10) &= 1 \\
  a_1(11) &= 1 \\
  b_2(00) &= 0 \\
  b_2(01) &= 0 \\
  b_2(10) &= 0 \\
  b_2(11) &= 1 \\
  b_3(00) &= 1 \\
  b_3(01) &= 0 \\
  b_3(10) &= 0 \\
  b_3(11) &= 0 \\
  a_4(00) &= 0 \\
  a_4(01) &= 0 \\
  a_4(10) &= 0 \\
  a_4(11) &= 1
\end{align*}
\]

A path tells us the bits sent
Protocol Trees

- We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

A path tells us the bits sent

so the height of the tree is the cost of the protocol
Protocol Trees

- We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

<table>
<thead>
<tr>
<th></th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

For a given $f$ we want to find the tree with the smallest height.
Protocol Trees

- We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

For a given $f$ we want to find the tree with the smallest height

(the $\mathcal{P}$ which uses the fewest bits)
Protocol Trees

• We now give a formal definition of a protocol $\mathcal{P}$ as a binary tree...

For a given $f$ we want to find the tree with the smallest height

$(the \mathcal{P} \text{ which uses the fewest bits})$

$D(f)$ (the communication complexity of $f$) is the height of the shortest tree
# Rectangles

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
the **EQUALITY** function

## Rectangles

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
## Rectangles

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
For any function \( f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\} \) we let \( M_f \) be the \( 2^n \times 2^n \) matrix where \( M_f[x, y] \) is the value of \( f(x, y) \)
Rectangles

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

- For any function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x,y]$ is the value of $f(x,y)$

- A rectangle is a subset of the cells of $M_f$ of the form $A \times B$
  
  Where $A$ is a subset of the rows and $B$ is a subset of the columns
Rectangles

- A rectangle is a subset of the cells of $M_f$ of the form $A \times B$
  
  Where $A$ is a subset of the rows and $B$ is a subset of the columns

- For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$
Rectangles

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

- For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$

- A rectangle is a subset of the cells of $M_f$ of the form $A \times B$
  Where $A$ is a subset of the rows and $B$ is a subset of the columns
### Rectangles

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

- For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$

- A *rectangle* is a subset of the cells of $M_f$ of the form $A \times B$
  Where $A$ is a subset of the rows and $B$ is a subset of the columns
Rectangles

For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$.

A rectangle is a subset of the cells of $M_f$ of the form $A \times B$ where $A$ is a subset of the rows and $B$ is a subset of the columns.
Rectangles

• For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$

• A rectangle is a subset of the cells of $M_f$ of the form $A \times B$
  Where $A$ is a subset of the rows and $B$ is a subset of the columns
For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$

A rectangle is a subset of the cells of $M_f$ of the form $A \times B$

Where $A$ is a subset of the rows and $B$ is a subset of the columns

A rectangle is monocromatic if all cells have the same value
• A rectangle is a subset of the cells of $M_f$ of the form $A \times B$
  Where $A$ is a subset of the rows and $B$ is a subset of the columns

• A rectangle is monocromatic if all cells have the same value

• For any function $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ we let $M_f$ be the $2^n \times 2^n$ matrix where $M_f[x, y]$ is the value of $f(x, y)$
Rectangles and Protocol trees

\[
\begin{align*}
  a_1(00) &= 0 \\
  a_1(01) &= 0 \\
  a_1(10) &= 1 \\
  a_1(11) &= 1 \\
  b_2(00) &= 0 \\
  b_2(01) &= 0 \\
  b_2(10) &= 0 \\
  b_2(11) &= 1 \\
  b_3(00) &= 1 \\
  b_3(01) &= 0 \\
  b_3(10) &= 0 \\
  b_3(11) &= 0 \\
  a_4(00) &= 0 \\
  a_4(01) &= 0 \\
  a_4(10) &= 0 \\
  a_4(11) &= 1
\end{align*}
\]

\[
\begin{array}{c|cccc}
  f & 00 & 01 & 10 & 11 \\
  \hline
  00 & 0 & 0 & 0 & 1 \\
  01 & 0 & 0 & 0 & 1 \\
  10 & 0 & 0 & 0 & 0 \\
  11 & 0 & 1 & 1 & 1 \\
\end{array}
\]
Consider which inputs are possible after messages $m_1, m_2 \ldots m_i$ have occurred.
Rectangles and Protocol trees

Consider which inputs are possible after messages $m_1, m_2 \ldots m_i$ have occurred.
Consider which inputs are possible after messages $m_1, m_2 \ldots m_i$ have occurred.
Rectangles and Protocol trees

Consider which inputs are possible after messages \( m_1, m_2 \ldots m_i \) have occurred.
Rectangles and Protocol trees

- Consider which inputs are possible after messages \( m_1, m_2 \ldots m_i \) have occurred
Rectangles and Protocol trees

• Consider which inputs are possible after messages $m_1, m_2 \ldots m_i$ have occurred

• The inputs $(x, y)$ which are possible after $m_1, m_2 \ldots m_i$ have been sent must form a rectangle
Consider which inputs are possible after messages \( m_1, m_2 \ldots m_i \) have occurred.

The inputs \((x, y)\) which are possible after \( m_1, m_2 \ldots m_i \) have been sent must form a rectangle.

If \( m_1, m_2 \ldots m_i \) is a root-leaf path the rectangle must be monocromatic.
Consider which inputs are possible after messages $m_1, m_2 \ldots m_i$ have occurred.

The inputs $(x, y)$ which are possible after $m_1, m_2 \ldots m_i$ have been sent must form a rectangle.

If $m_1, m_2 \ldots m_i$ is a root-leaf path the rectangle must be monocromatic.
Rectangles and Protocol trees

Let $C(f)$ be the minimum number of monocromatic rectangles which partition $M_f$. 

<table>
<thead>
<tr>
<th>$f$</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Rectangles and Protocol trees

Let $C(f)$ be the minimum number of monocromatic rectangles which partition $M_f$.

(in this case 5)
Rectangles and Protocol trees

\[a_1(00) = 0\]
\[a_1(01) = 0\]
\[a_1(10) = 1\]
\[a_1(11) = 1\]

\[b_2(00) = 0\]
\[b_2(01) = 0\]
\[b_2(10) = 0\]
\[b_2(11) = 1\]

\[b_3(00) = 1\]
\[b_3(01) = 0\]
\[b_3(10) = 0\]
\[b_3(11) = 0\]

\[a_4(00) = 0\]
\[a_4(01) = 0\]
\[a_4(10) = 0\]
\[a_4(11) = 1\]

\[y \text{ (Bob)}\]

\[
\begin{array}{c|cccc}
f & 00 & 01 & 10 & 11 \\
\hline
00 & 0 & 0 & 0 & 1 \\
01 & 0 & 0 & 0 & 1 \\
10 & 0 & 0 & 0 & 0 \\
11 & 0 & 1 & 1 & 1 \\
\end{array}
\]

\[x\]

\[\bullet \text{ Let } C(f) \text{ be the minimum number of monocromatic rectangles which partition } M_f (in this case 5)\]

\[\bullet \text{ No correct protocol tree for } f \text{ can have fewer than } C(f) \text{ leaves}\]
Rectangles and Protocol trees

Let $C(f)$ be the minimum number of monocromatic rectangles which partition $M_f$ (in this case 5).

- No correct protocol tree for $f$ can have fewer than $C(f)$ leaves.
- So no protocol for $f$ can use fewer than $\log_2 C(f)$ bits.
Let $C(f)$ be the minimum number of monocromatic rectangles which partition $M_f$ (in this case 5).

- No correct protocol tree for $f$ can have fewer than $C(f)$ leaves.
- So no protocol for $f$ can use fewer than $\log_2 C(f)$ bits.

That is $D(f) \geq \log_2 C(f)$.
There is a naive $n + 1$ bit protocol

Alice and Bob want to compute $f(x, y) = 1$ iff $x = y$

There is a naive $n + 1$ bit protocol

We will now prove that $D(EQ) \geq n$

i.e. there is no protocol which always takes less than $n$ bits
**QUALITY**

**E**

What is $C(EQ)$, the minimum number of monocromatic rectangles we can partition $M_{EQ}$ in to?
What is \( C(EQ) \), the minimum number of monocromatic rectangles we can partition \( M_{EQ} \) in to?

Consider any rectangle \( A \times B \) that includes \( M_f[i, i] \) and \( M_f[j, j] \) \((i \neq j)\).
**EQUALITY**

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

- What is $C(EQ)$, the minimum number of monocromatic rectangles we can partition $M_{EQ}$ in to?

- Consider any rectangle $A \times B$ that includes $M_f[i, i]$ and $M_f[j, j]$
  - We have that $i, j \in A$ and $i, j \in B$ \hspace{1cm} (i \neq j)
• What is $C(EQ)$, the minimum number of monocromatic rectangles we can partition $M_{EQ}$ in to?

• Consider any rectangle $A \times B$ that includes $M_f[i, i]$ and $M_f[j, j]$
  
  ○ We have that $i, j \in A$ and $i, j \in B$  
  
  ○ So $M_f[i, j] = 0$ and $M_f[j, i] = 0$ are in $A \times B$

  Hence $A \times B$ is not monocromatic
• What is $C(\text{EQ})$, the minimum number of monocromatic rectangles we can partition $M_{\text{EQ}}$ into?

• Consider any rectangle $A \times B$ that includes $M_f[i, i]$ and $M_f[j, j]$
  
  o We have that $i, j \in A$ and $i, j \in B$  \hspace{1cm} (i \neq j)
  
  o So $M_f[i, j] = 0$ and $M_f[j, i] = 0$ are in $A \times B$

  Hence $A \times B$ is not monocromatic

• So every $M_f[i, i]$ must be in a different rectangle under any partition
**EQUALITY**

<table>
<thead>
<tr>
<th>EQ</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

- What is $C(EQ)$, the minimum number of monocromatic rectangles we can partition $M_{EQ}$ into?

- So every $M_f[i, i]$ must be in a different rectangle under any partition.
What is $C(\text{EQ})$, the minimum number of monocromatic rectangles we can partition $M_{\text{EQ}}$ in to?

So every $M_f[i, i]$ must be in a different rectangle under any partition.

As there are $2^n$ cells of the form $M_f[i, i]$ there must be at least $2^n$ rectangles in any partition.
• What is $C(EQ)$, the minimum number of monocromatic rectangles we can partition $M_{EQ}$ in to?

• So every $M_f[i, i]$ must be in a different rectangle under any partition

• As there are $2^n$ cells of the form $M_f[i, i]$ there must be at least $2^n$ rectangles in any partition

• Therefore $C(EQ) \geq 2^n$ and hence

$$D(EQ) \geq \log_2 C(EQ) \geq n$$
The **DISJOINTNESS** problem

\[ x = x_1 x_2 x_3 \ldots x_n \quad \text{and} \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute \( \text{Dis}(x, y) \)

  \[ \text{Dis}(x, y) = 1 \] if there is no index, \( i \), such that \( x_i = y_i = 1 \)

  (and \( \text{Dis}(x, y) = 0 \) otherwise)
The **Disjointness** problem

\[ x = x_1x_2x_3 \ldots x_n \quad y = y_1y_2y_3 \ldots y_n \]

- Alice and Bob want to compute \( \text{Dis}(x, y) \)

  \[ \text{Dis}(x, y) = 1 \text{ if there is no index, } i, \text{ such that } x_i = y_i = 1 \]

  (and \( \text{Dis}(x, y) = 0 \) otherwise)

So \( \text{Dis}(0110, 0001) = 1 \), \( \text{Dis}(0110, 0011) = 0 \) and \( \text{Dis}(0110, 0100) = 0 \)
The **Disjointness** problem

\[ x = x_1 x_2 x_3 \ldots x_n \quad \text{and} \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute \( \text{Dis}(x, y) \)

  \[ \text{Dis}(x, y) = 1 \text{ if there is no index, } i, \text{ such that } x_i = y_i = 1 \]
  
  (and \( \text{Dis}(x, y) = 0 \) otherwise)

So \( \text{Dis}(0110, 0001) = 1 \), \( \text{Dis}(0110, 0011) = 0 \) and \( \text{Dis}(0110, 0100) = 0 \)

- For **Disjointness**, we will show that \( n \leq D(\text{Dis}) \leq n + 1 \)

  *i.e. there is no protocol which always takes fewer than \( n \) bits*
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>Dis</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>DIS</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is \( C(\text{Dis}) \), the minimum number of monocromatic rectangles we can partition \( M_{\text{Dis}} \) in to?

- Consider any rectangle \( A \times B \) that includes \( M_f[g, \overline{g}] = M_f[h, \overline{h}] = 1 \) (for any \( g \neq h \))
The **Disjointness** problem

<table>
<thead>
<tr>
<th>Dis</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?

- Consider any rectangle $A \times B$ that includes $M_f[g, \overline{g}] = M_f[h, \overline{h}] = 1$ (for any $g \neq h$)

$\overline{g}$ is the negation of $g$
The DISJOINTNESS problem

<table>
<thead>
<tr>
<th>DIS</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?

- Consider any rectangle $A \times B$ that includes $M_f[g, \overline{g}] = M_f[h, \overline{h}] = 1$
  - So $M_f[g, \overline{h}]$ and $M_f[h, \overline{g}]$ are in $A \times B$ (for any $g \neq h$)

$\overline{g}$ is the negation of $g$
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>Dis</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?

- Consider any rectangle $A \times B$ that includes $M_f[g, \overline{g}] = M_f[h, \overline{h}] = 1$
  
  - So $M_f[g, \overline{h}]$ and $M_f[h, \overline{g}]$ are in $A \times B$ (for any $g \neq h$)
  
  - As $g \neq h$, there exists $g_i \neq h_i$. 

$\overline{g}$ is the negation of $g$
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>DIS</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{DIS})$, the minimum number of monocromatic rectangles we can partition $M_{\text{DIS}}$ in to?

- Consider any rectangle $A \times B$ that includes $M_f [g, \overline{g}] = M_f [h, \overline{h}] = 1$
  - So $M_f [g, \overline{h}]$ and $M_f [h, \overline{g}]$ are in $A \times B$ (for any $g \neq h$)
  - As $g \neq h$, there exists $g_i \neq h_i$.
    - If $g_i = 1$ then $\overline{h_i} = 1$ and $M_f [g, \overline{h}] = 0$

$\overline{g}$ is the negation of $g$
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>Dis</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $\mathcal{C}(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?

- Consider any rectangle $A \times B$ that includes $M_f[g, \overline{g}] = M_f[h, \overline{h}] = 1$
  - So $M_f[g, \overline{h}]$ and $M_f[h, \overline{g}]$ are in $A \times B$
  - As $g \neq h$, there exists $g_i \neq h_i$.
    - If $g_i = 1$ then $\overline{h_i} = 1$ and $M_f[g, \overline{h}] = 0$
    - If $g_i = 0$ then $\overline{g_i} = 1$, $h_i = 1$ and $M_f[h, \overline{g}] = 0$

$\overline{g}$ is the negation of $g$
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>DIS</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is \( C(\text{DIS}) \), the minimum number of monocromatic rectangles we can partition \( M_{\text{DIS}} \) in to?

- Consider any rectangle \( A \times B \) that includes \( M_f[g, \overline{g}] = M_f[h, \overline{h}] = 1 \)
  
  - So \( M_f[g, \overline{h}] \) and \( M_f[h, \overline{g}] \) are in \( A \times B \) (for any \( g \neq h \))
  
  - As \( g \neq h \), there exists \( g_i \neq h_i \).

    - If \( g_i = 1 \) then \( \overline{h_i} = 1 \) and \( M_f[g, \overline{h}] = 0 \)
    - If \( g_i = 0 \) then \( \overline{g_i} = 1, \overline{h_i} = 1 \) and \( M_f[h, \overline{g}] = 0 \)

Hence \( A \times B \) is not monocromatic
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>Dis</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

• **What is $C(\text{Dis})$, the minimum number of monocromatic rectangles** we can partition $M_{\text{Dis}}$ in to?

• So every $M_f[g, \overline{g}]$ must be in a different rectangle under any partition

\[ \overline{g} \text{ is the negation of } g \]
The **Disjointness** problem

<table>
<thead>
<tr>
<th></th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?

- So every $M_f[g, \bar{g}]$ must be in a different rectangle under any partition

- As there are $2^n$ cells of the form $M_f[g, \bar{g}]$, there must be at least $2^n$ rectangles in any partition

$\bar{g}$ is the negation of $g$
The **DISJOINTNESS** problem

<table>
<thead>
<tr>
<th>DIS</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>010</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- What is $C(\text{Dis})$, the minimum number of monocromatic rectangles we can partition $M_{\text{Dis}}$ in to?

- So every $M_f[g, \bar{g}]$ must be in a different rectangle under any partition.

- As there are $2^n$ cells of the form $M_f[g, \bar{g}]$ there must be at least $2^n$ rectangles in any partition.

- Therefore $C(\text{Dis}) \geq 2^n$ and hence $D(\text{EQ}) \geq \log_2 C(\text{Dis}) \geq n$.
The **INNERPRODUCT** problem

\[ x = x_1 x_2 x_3 \ldots x_n \quad \text{and} \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute \( \text{IP}(x, y) \)

\[
\text{IP}(x, y) = \sum_{i=1}^{n} (x_i \cdot y_i) \mod 2
\]
The **INNERPRODUCT** problem

\[ x = x_1 x_2 x_3 \ldots x_n \quad \text{and} \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute \( \text{IP}(x, y) \)

\[ \text{IP}(x, y) = \sum_{i=1}^{n} (x_i \cdot y_i) \mod 2 \]

\[ \text{IP}(0110, 0011) = (0 \cdot 0) + (1 \cdot 0) + (1 \cdot 1) + (0 \cdot 1) \mod 2 = 1 \mod 2 \]
The **INNERPRODUCT** problem

\[ x = x_1 x_2 x_3 \ldots x_n \quad y = y_1 y_2 y_3 \ldots y_n \]

- Alice and Bob want to compute \( \text{IP}(x, y) \)

\[
\text{IP}(x, y) = \sum_{i=1}^{n} (x_i \cdot y_i) \mod 2
\]

\[
\text{IP}(0110, 0011) = (0 \cdot 0) + (1 \cdot 0) + (1 \cdot 1) + (0 \cdot 1) \mod 2 = 1 \mod 2
\]

For **INNERPRODUCT**, it can be shown that \( n \leq D(\text{IP}) \leq n + 1 \)

(but we don’t have time for the proof)
Conclusions

- The communication complexity of \( f \), \( D(f) \) is the minimum cost of any protocol \( \mathcal{P} \), which correctly computes \( f \).
- It’s the minimum number of bits of communication Alice and Bob need to communicate to compute \( f(x, y) \) (for any \( (x, y) \)).

- Any protocol for the \textsc{Equality} problem requires \( \geq n \) bits
  
  i.e. \( D(\text{Equality}) \geq n \) bits

- Any protocol for the \textsc{Disjointness} problem requires \( \geq n \) bits
  
  i.e. \( D(\text{Disjointness}) \geq n \) bits

- Any protocol for the \textsc{InnerProduct} problem requires \( \geq n \) bits
  
  i.e. \( D(\text{InnerProduct}) \geq n \) bits