Definition: An Interactive proof protocol is given by two functions:

\[ V : \Sigma^* \times \Sigma^* \times \Sigma^* \to \Sigma^* \cup \{ \text{accept}, \text{reject} \} \]
\[ P : \Sigma^* \to \Sigma^* \]

Let \( s_i \) denote the concatenation of \( i \) pairs of messages, \( s_i = \#x_1 \#y_1 \# \cdots \#x_i \#y_i \). We write \( V(w, r, s_i) = x_{i+1} \) to mean that \( V \) on input \( w \), with random sequence \( r \), and current message stream \( s \) produces next message \( x_{i+1} \). We say \( P(s_i \# x_{i+1}) = y_{i+1} \) to mean that \( P \) produces next message \( y_{i+1} \) given current message stream \( s_i \# x_{i+1} \). The exchange of a single pair of messages is called a round.
For a given input $w$ and random sequence $r$ we say

$$(V*P)(w,r) \text{ accepts}$$

if there exists a message stream $s = \#x_1\#y_1\# \cdots \#x_l\#y_l$ such that $V(w,r,s) = \text{accept}$, and for each $i < l$, $V(w,r,s_i) = x_{i+1}$ and $P(s_i\#x_{i+1}) = y_{i+1}$. 
Let us assume for simplicity that there is a function $l$ such that for inputs $w$ of length $n$, $V$ will only accept if the length of $r$ is $l(n)$. Then we write

$$\Pr[(V \ast P)(w) \text{ accepts}]$$

to mean $\Pr[(V \ast P)(w, r) \text{ accepts}]$ for $r$ chosen randomly from $\Sigma^{\lfloor|w|\rfloor}$.

Further we let

$$\Pr[V(w) \text{ accepts}]$$

denote $\max_p \Pr[(V \ast P)(w) \text{ accepts}]$. 
Let the language of the verifier, \( L(V) = \{ w: \Pr[V(w) \text{ accepts}] > 1/2 \} \)

Say \( V \) has error probability \( e \) if for all \( w \in \Sigma^* \):

1) if \( w \in L(V) \), \( \Pr[V(w) \text{ accepts}] \geq 1 - e \)
2) if \( w \notin L(V) \), \( \Pr[V(w) \text{ accepts}] \leq e \)

For \( W \subseteq \Sigma^* \), we say \( W \in \text{IP} \) if there is a polynomial time verifier \( V \) with error probability 1/3 accepting \( W \). As we shall see later, the class \( \text{IP} \) is unaffected if we substitute \( e \) for 1/3, where \( 2^{-\text{poly}(n)} \leq e \leq 1/2 - 2^{-\text{poly}(n)} \).
Definition: An Interactive proof protocol with public coin is defined as above with the following difference. The random input $r$ is considered to be the concatenation of $l$ strings $r = r_1 r_2 \cdots r_l$ where $l$ is the number of rounds and $V$ is restricted to produce $r_i$ as it’s $i^{th}$ message, i.e., for $i \leq l$, $V(w, r, s_i) = r_i$ or accept or reject.
This notion is essentially identical to that of the Arthur-Merlin game defined by Babai in [B]. Following his terminology we say that for $W \subseteq \Sigma^*$, $W \in \text{AM}(\text{poly})$ if $W \in \text{IP}$ as above and the interactive proof protocol uses a public coin. We refer to an Arthur-Merlin game as an $A-M$ protocol.

For polynomial $Q$, say $W \in \text{IP}[Q(n)]$ if $W \in \text{IP}$ with a verifier which never sends more than $Q(n)$ messages for inputs of length $n$. Similarly define $\text{AM}[Q(n)]$. 
4.1. Approximate lower bound lemma

This lemma, an application of Carter-Wegman universal hashing [CW], due to Sipser [Si], plays a key role in our proof of equivalence. Its application to approximate lower bounds was first given by Stockmeyer [St]. Its application in Arthur-Merlin protocols first appears in Babai [B].
Definition: Let $D$ be a $k \times b$ Boolean matrix. The linear function $h_D: \Sigma^k \rightarrow \Sigma^b$ is given by $h_D(x) = x \cdot D$ using ordinary matrix multiplication modulo 2. A random linear function is obtained by selecting the matrix $D$ at random. If $H = \{h_1, \ldots, h_l\}$ is a collection of functions, $C \subseteq \Sigma^k$, and $D \subseteq \Sigma^b$ then $H(C)$ denotes $\bigcup h_i(C)$, and $H^{-1}(D)$ denotes $\bigcup h_i^{-1}(D)$. Let $|C|$ denote the cardinality of $C$. 
Lemma: Given $b,k,l > 0$, $l > \max(b,8)$, and $C \subseteq \Sigma^k$. Randomly select $l$ linear functions $H = \{h_1, \ldots, h_l\}$, $h_i : \Sigma^k \to \Sigma^b$ and $l^2$ strings $Z = \{z_1, \ldots, z_{l^2}\} \subseteq \Sigma^b$. Then

1. If $b = 2 + \lceil \log |C| \rceil$ then
   a) $\Pr[|H(C)| \geq |C|/l] \geq 1 - 2^{-l}$
   b) $\Pr[H(C) \cap Z \neq \emptyset] \geq 1 - 2^{-l/8}$

2.
   a) $|H(C)| \leq l|C|
   b) If for $d > 0$, $|C| \leq 2^b/d$ then:
      $\Pr[H(C) \cap Z \neq \emptyset] \leq l^3/d$

**if** $2^b/4 \geq |C| \geq 2^b/8$ **then** $\Pr[H(C) \cap Z = \emptyset] \leq 2^{-l/8}$

**if** $|C| \leq 2^b/d$, $d > 0$, **then** $\Pr[H(C) \cap Z \neq \emptyset] \leq l^3/d$
1. If \( b = 2 + \lceil \log |C| \rceil \) then
   a) \( \Pr[|H(C)| \geq |C|/l] \geq 1 - 2^{-l} \)
   b) \( \Pr[H(C) \cap Z \neq \emptyset] \geq 1 - 2^{-l/8} \)

**Proof 1a:** Since \( 2^b \geq 4|C| \) the following chain of statements are easily verified. Let \( (h_i(x))^j \) denote the \( j \)th bit of the string \( h_i(x) \). Fix \( x,y \in \Sigma^k, x \neq y, i,j > 0 \), except where quantified.

\[
\Pr[(h_i(x))^j = (h_i(y))^j] = 1/2
\]

\[
\Pr[h_i(x) = h_i(y)] = 2^{-b}
\]

\[
\Pr[\exists y \in C \ (x \neq y \& h_i(x) = h_i(y))] \leq |C| \cdot 2^{-b} \leq 1/4
\]

\[
\Pr[\forall i \leq l \ \exists y \in C \ (x \neq y \& h_i(x) = h_i(y))] \leq 4^{-l}
\]

\[
\Pr[\exists x \in C \ \forall i \leq l \ \exists y \in C \ (x \neq y \& h_i(x) = h_i(y))] \leq |C| \cdot 4^{-l} \leq 2^{-l}
\]

Therefore \( \Pr[|H(C)| \geq |C|/l] \geq 1 - 2^{-l} \)
1. If $b = 2 + \lfloor \log |C| \rfloor$ then
   a) $\Pr[|H(C)| \geq |C|/l] \geq 1 - 2^{-l}$
   b) $\Pr[H(C) \cap Z \neq \emptyset] \geq 1 - 2^{-l/8}$

Proof 1b: Since $|C| \geq 2^b/8$, if $|H(C)| \geq |C|/l$ then

$$\frac{|H(C)|}{|\Sigma^b|} \geq \frac{1}{8l}$$

Thus it is likely that one of the $l^2$ strings in $Z$ will be in $H(C)$.

$\Pr[H(C) \cap Z = \emptyset] \leq (1 - 1/8l)^{l^2} + 2^{-l} < 2^{-l/8}$
2.

a) \(|H(C)| \leq l|C|\)

b) If for \(d > 0\), \(|C| \leq 2^b/d\) then:
\[\Pr[H(C) \cap Z \neq \emptyset] \leq l^3/d\]

**Proof 2a:** Obvious.

**Proof 2b:** Since
\[
\frac{|H(C)|}{\Sigma^b} \leq \frac{l|C|}{d|C|} = \frac{l}{d}
\]

The probability that each \(z_i\) is in \(H(C)\) is at most \(l/d\). Thus the probability that any of the \(l^2\) strings in \(Z\) is in \(H(C)\) is at most \(l^3/d\).
We use this lemma to obtain Arthur-Merlin protocols for showing an approximate lower bound on the size of sets. Let $C$ be a set in which Arthur can verify membership, possibly with Merlin's help. Then let Arthur picks random $H$ and $Z$ and Merlin attempt to respond with $x \in C$ such that some $x \in H^{-1}(z)$. If $C$ is large then he will likely succeed and if $C$ is small he will likely fail.

If $2^b/4 \geq |C| \geq 2^b/8$ then $\Pr[H(C) \cap Z = \emptyset] \leq 2^{-l/8}$

If $|C| \leq 2^b/d$, $d > 0$, then $\Pr[H(C) \cap Z \neq \emptyset] \leq l^3/d$
4.2. Main Theorem

Theorem: $\text{IP}[Q(n)] = \text{AM}[Q(n) + 2]$ for any polynomial $Q(n)$
An informal proof sketch: Let’s focus on 1-round protocols. Assume \( V \) has an exponentially small error probability \( e \), sends only messages of length \( m \), and uses random sequences of length \( l \). For each \( x \in \Sigma^m \) let 
\[
\beta_x = \{ r : V(r,w,#) = x \}.
\]
For every \( y \in \Sigma^m \) let 
\[
\alpha_{xy} = \{ r : r \in \beta_x \quad \& \quad V(r,w,#x#y) = \text{accept} \}.
\]
Clearly, for each \( x \), the optimal prover will select a \( y_x \) maximizing \( |\alpha_{xy}| \). Let \( \alpha_x = \alpha_{xy} \). Let \( \alpha_0 = \bigcup_x \alpha_x \). Then 
\[
\Pr[V(w) \text{ accepts}] = \frac{|\alpha_0|}{2^l}.
\]
We next present the protocol by which $A$ and $M$ simulate $V$ and $P$. $M$ tries to convince $A$ that $|\alpha_0| > e \cdot 2^l$ because this implies that $\Pr[V(w) \text{ accepts}] > e$ and hence $\approx 1$. He does this by showing that there are "many" $\alpha_x$'s which are "large", where "many" $\times$ "large" $> e \cdot 2^l$. The tradeoff between "many" and "large" is governed by a parameter $b$ sent by $M$ to $A$. 
More precisely, $M$ first sends $b$ to $A$. Then two approximate lower bound protocols ensue. The first convinces $A$ that $|\{x: |\alpha_x| \geq \frac{2b}{(e \cdot 2^l)}\}| \geq 2^b$. $M$ produces an $x$ in that set as per the approximate lower bound lemma. The second convinces $A$ that $x$ really is in that set as claimed, i.e., that $|\alpha_x| \geq \frac{2b}{(e \cdot 2^l)}$. 

$e \frac{l}{2^b}$

$e \frac{l}{2^b}$
For $g$-round protocols iterate the first approximate lower bound protocol to obtain $\alpha_0 \supseteq \alpha_1 \supseteq \cdots \supseteq \alpha_g$ where there are "many,$i$" ways to extend $\alpha_{i-1}$ to $\alpha_i$ and $\alpha_g$ is "large". Require that $(\prod"many_i") \times "large" \geq e \cdot 2^l$. 
Full proof: Let $W \in \text{IP}[Q(n)]$. We may assume, without loss of generality, that on inputs $w$ of length $n$ there are exactly $g(n) = Q(n)/2$ pairs of messages sent between $V$ and $P$, these messages are exactly $m(n)$ long and the random input $r$ to $V$ is $l(n)$ long. Let $e(n)$ bound the error probability.
Amplification Lemma: Let $p(n)$ be a polynomial. Let $V$ be a verifier which on inputs of length $n$ a total of at most $g(n)$ messages, each of length $m(n)$, using $l(n)$ random bits, and with error probability at most $1/3$. Then there is a $V'$ such that $L(V) = L(V')$, with a total of at most $g(n)$ messages, each of length $O(p(n)m(n))$, using $O(p(n)l(n))$ random bits and with an error probability of at most $2^{-p(n)}$.

proof: $V'$ performs $O(p(n))$ independent parallel simulations of $V$ and takes the majority vote of the outcomes. Details left to the reader. ■
Let $X = \sum_i X_i$ be a sum of independent random indicator variables $X_i$. For each $i$, let $p_i = \Pr[X_i = 1]$, and let $\mu = \mathbb{E}[X] = \sum_i \mathbb{E}[X_i] = \sum_i p_i$.

**Chernoff Bound (Upper Tail).**

$$\Pr[X > (1 + \delta)\mu] < \left( \frac{e^\delta}{(1 + \delta)^{1+\delta}} \right)^\mu \text{ for any } \delta > 0.$$

\[ \Pr[X > (1 + \delta)\mu] < e^{-\mu\delta^2/3} \] for any $0 < \delta < 1$.

**Chernoff Bound (Lower Tail).**

$$\Pr[X < (1 - \delta)\mu] < \left( \frac{e^\delta}{(1 - \delta)^{1-\delta}} \right)^\mu < e^{-\mu\delta^2/2} \text{ for any } \delta > 0.$$
By this lemma we may assume:

\[ e(n) \leq l(n)^{-12g^2(n)} \]

Further we may assume that
\[ l(n) > \max(g(n), m(n), 80) \].
We write \( g, m, e, l \) for \( g(n), m(n), e(n) \) and \( l(n) \) where \( n \) is understood.
We now describe the functions $A$ and $M$, simulating $V$ and $P$, informally as two parties exchanging messages. The variables $x_i$ and $y_i$ represent messages sent by $V$ and $P$ respectively. In essence, the idea is for $A$ to use the random hash functions to force $M$ to produce a generic run of the $V,P$ protocol and then finally to prove that this run would likely cause $V$ to accept. The numbers $b_i$ that $M$ produces roughly correspond to the log of the number of possible generic messages that $V$ can make at round $i$. 
Arthur-Merlin Games

\( w \in L \)

\( b_1, H, Z \)

1 \( \leq i \leq g \), \( V(w, r, s_{i-1}) = x_i \) ?

\( x_i \in H^{-1}(Z) \) ?

\( r \in H^{-1}(Z) ? \)

\( 1 \leq i \leq g \), \( V(w, r, s_g) = "accept" \)

\( \sum b_i \geq l - g \log l ? \)
Poly-Time

NOTA: $[x]=\text{ceiling}(x)$

$b_1 = 2 + \lceil \log |\gamma_{\text{max}}| \rceil$

$H \in \mathbb{R}(\sum m \rightarrow \sum^{b_1})$, $Z \in \mathbb{R}(\sum^m)^2$

$x_i \in H^{-1}(Z)$, $\alpha_{x_i} \in \gamma_{\text{max}}$, $y_i = \text{P}(s_{i-1} \# x_i)$, $b_{i+1} = 2 + \lceil \log |\gamma_{\text{max}}| \rceil$

$1 \leq i \leq g$

$H \in \mathbb{R}(\sum^m \rightarrow \sum^{b_{i+1}})$, $Z \in \mathbb{R}(\sum^m)^2$

$r \in H^{-1}(Z)$?

$1 \leq i \leq g$, $V(w, r, s_{i-1}) = x_i$?

$V(w, r, s_g) = \text{“accept”}$?

$\sum b_i \geq l - g \log l$?
Round 0:

A initially makes a null move and receives number $b_1$ from $M$. Go to round 1.

Round $i$ ($1 \leq i \leq g$):

So far $A$ has received $b_1, \ldots, b_i$, and strings $x_1, \ldots, x_{i-1}, y_1, \ldots, y_{i-1}$ from $M$. Now $A$ randomly selects $l$ linear functions $H = \{h_1, \ldots, h_l\}$, $h_i : \Sigma^m \rightarrow \Sigma^{b+1}$ and $l^2$ strings $Z = \{z_1, \ldots, z_{l^2}\} \subseteq \Sigma^{b+1}$ and then sends to $M$. $A$ then expects to receive strings $x_i$ and $y_i$ and number $b_{i+1}$ from $M$. $A$ checks that $x_i \in H^{-1}(Z)$. If not then $A$ immediately rejects. Then $A$ performs round $i+1$.

Final round $g+1$:

Let $s_i = x_1 \# y_1 \# \cdots \# x_i \# y_i$. $A$ randomly selects $l$ linear functions $H = \{h_1, \ldots, h_l\}$, $h_i : \Sigma^l \rightarrow \Sigma^{b_g+1}$ and $l^2$ strings $Z \subseteq \Sigma^{b_g+1}$. It then expects to receive a string $r \in \Sigma^l$ from $M$ and checks that $r \in H^{-1}(Z)$. $A$ accepts if for each $i \leq g$ $V(w, r, s_i) = x_{i+1}$, $V(w, r, s_g) = \text{accept}$ and $\sum b_i \geq l - g \log l$. (and sends to $M$)
Can Merlin convince Arthur?

Now we show that \( \Pr[V(w) \text{ accepts}] > e(n) \) iff \( \Pr[A(w) \text{ accepts}] \geq 2/3 \).

\((\rightarrow)\) Merlin’s protocol when \( w \in W \)

First some notation. For \( r \in \Sigma^l \) and \( s = v_1 \# v_2 \# \cdots \# v_k \) a stream of messages we say

\((V*P)(w,r) \text{ accepts via } s\)

if the first \( k \) messages sent by \( V \) and \( P \) agree with \( s \) and \( (V*P)(w,r) \text{ accepts} \).

Suppose \( \Pr[V(w) \text{ accepts}] \geq 2/3 \). Fix any \( P \) such that \( \Pr[(V*P)(w) \text{ accepts}] \geq 2/3 \). We now exhibit a protocol for \( M \) such that \( \Pr[(A*M)(w) \text{ accepts}] \geq 2/3 \).
Obtain \( b_i \) \((i \leq g)\): Let 
\[ s_{i-1} = \#x_1\#y_1\# \cdots \#x_{i-1}\#y_{i-1} \] 
be the message stream for the V-P protocol produced so far. For each \( x \in \Sigma^m \) let \( \alpha_x = \{ r : (V*P)(w,r) \) accepts via \( s_{i-1}\#x \} \). Group these \( \alpha \)'s into \( l \) classes \( \gamma_1, \ldots, \gamma_l \) where \( \gamma_d \) contains \( \alpha \)'s of size \( > 2^{d-1} \) and \( \leq 2^d \). Choose the class \( \gamma_{\text{max}} \) whose union \( \bigcup \gamma_{\text{max}} = \bigcup \{ \alpha_x : \alpha_x \in \gamma_{\text{max}} \} \) is largest. Send \( b_i = 2 + \lceil \log |\gamma_{\text{max}}| \rceil \).

Round \( i \):

\( M \) receives \( h_1, \ldots, h_l \) from \( A \) and strings \( z_1, \ldots, z_l \). If there is an \( x \in H^{-1}(Z) \) such that \( \alpha_x \in \gamma_{\text{max}} \), call it \( x_i \). Then, \( M \) responds with the pair \( x_i, y_i \) where 
\[ y_i = P(s_{i-1}\#x_i) \]. Otherwise \( M \) responds with "failure". In the later analysis we refer to the set \( \alpha_{x_i} \) as \( \alpha_i \). Set \( i \leftarrow i + 1 \). Goto "obtain \( b_i \)."
Merlin’s protocol

Obtain $b_{g+1}$: $M$ produces the value $b_{g+1}$ as follows: Let $s_g = s_{g-1} \# y_{g} \# y_{g}$ be the message stream that has been selected. So $g_g = \text{stream}$. Send $b_{g+1} = 2 + \log |a|$. 

Round $g+1$:

$g_{g+1} = \{r: (V^P)(w, r) \}$ accepts via $s_g$. 

$M$ receives $h_1, \ldots, h_n$ and strings $r \in \mathbb{G}$ implies that $V(w, r, s) = \text{accept}$.

If there is an $\epsilon > 0$, $M$ responds with "failure". (Note that $\epsilon \leq \frac{1}{2}$.)

Else $\epsilon < \frac{1}{2}$, $M$ responds with "success". (Note that $\epsilon > \frac{1}{2}$.)

End of Protocol.
We now show that $\Pr[(A*M)(w) \text{ accepts}] \geq 2/3$. Let $\alpha_0 = \{r: (V*P)(w,r) = \text{ accept}\}$. Since $\Pr[V \text{ accepts } w]$ is high, $|\alpha_0| \geq (2/3)2^l$. By the definition of $M$, $A$ will accept provided $M$ never responds "failure" and $\sum b_i \geq l - g \log l$. By the approximate lower bound lemma the probability that $M$ responds failure at any round is $\leq 2^{-l/8}$. Hence, the probability that $M$ ever responds failure is $\leq g2^{-l/8} << 1/3$.

if $2^b/4 \geq |C| \geq 2^b/8$ then $\Pr[H(C) \cap Z = \emptyset] \leq 2^{-l/8}$

(Full proof $w \in L$)
The following two claims show that
\[ \sum b_i \geq l - g \log l. \]
Claim 1: For each \( 0 \leq i < g \)
\[ |\alpha_i| \geq \frac{|\alpha_{i-1}|}{l2^{b_i}} \]
Proof: Consider round \( i \) and the sets \( \alpha_x \) defined in "obtain \( b_i \)". By definition the \( \alpha_x \)'s partition \( \alpha_{i-1} \) and hence \( \bigcup \alpha_x = \alpha_{i-1} \). Hence

\[
|\bigcup \gamma_{\text{max}}| \geq \frac{|\alpha_{i-1}|}{l}
\]

there are \( l \) possibilities for \( \gamma_i \), thus at least one is of size total/\( l \).

Since all members of \( \gamma_{\text{max}} \) differ in size by at most a factor of 2 and since \( \alpha_i \in \gamma_{\text{max}} \) we have

\[
|\alpha_i| \geq \frac{|\bigcup \gamma_{\text{max}}|}{2|\gamma_{\text{max}}|}
\]

and since \( b_i = 2 + \lceil \log |\gamma_{\text{max}}| \rceil \) we have

\[
2^{b_{i+1}} \geq 2|\gamma_{\text{max}}|
\]

Thus

\[
|\alpha_i| \geq \frac{|\bigcup \gamma_{\text{max}}|}{2^{b_i}} \geq \frac{|\alpha_{i-1}|}{l2^{b_i}}
\]

( Claim 1/6 )

( Full proof \( w \in L \) )
Claim 2: $\sum b_i \geq l - g \log l$

Proof: By Claim 1 we have:

$$|\alpha_g| \geq \frac{|\alpha_0|}{l^g \cdot \prod_{i \leq g} 2^{b_i}}$$

Since $|\alpha_0| \geq (2/3)2^l$ and taking logs

$$\log |\alpha_g| \geq (l-1) - (g \log l + \sum_{i \leq g} b_i)$$

Since $b_{g+1} > 1 + \log |\alpha_g|$

$$\sum_{i \leq g+1} b_i \geq l - g \log l$$

$\square$

( Claim 2/6 )

( Full proof $w \in L$ )
Shafi Goldwasser, Michael Sipser

**Arthur-Merlin Games**

\[ w \notin L \]

**Poly-Time**

\[ b_1 \]

\[ H \in_R (\sum^m \rightarrow \sum^{b_1})^l, \ Z \in_R (\sum^m)^2 \]

**1 \leq i \leq g**

\[ x_i, y_i, b_{i+1} \]

\[ H \in_R (\sum^m \rightarrow \sum^{b_{i+1}})^l, \ Z \in_R (\sum^m)^2 \]

\[ r \]

\[ r \in H^{-1}(Z) \]

\[ 1 \leq i \leq g, \ V(w,r,s_{i-1}) = x_i ? \]

\[ V(w,r,s_g) = \text{“accept”} ? \]

\[ \sum b_i \geq l - g \log l ? \]
(←) Merlin's impotence when \( w \notin W \)

Show that if \( \Pr[V(w) \text{ accepts}] \leq \epsilon \),
then \( \Pr[A(w) \text{ accepts}] \leq 1/3 \).

For every \( i > 0 \) and
\[
s_i = x_1 \# y_1 \# \cdots \# x_i \# y_i
\]
let \( a(s_i) = \max_p \Pr[(V*P)(w) \text{ accepts via } s_i] \).
For each \( x \in \Sigma^m \)
let \( y_x \) be any \( y \in \Sigma^m \) maximizing \( a(s_i \# x \# y) \).
The following three claims show that $a(s_{i+1})$ is likely to be much smaller than $a(s_i)$.

Claim 3: $a(s_i) = \sum_x a(s_i \# x \# y_x)$
Fix $0 \leq i < g$ and $s_i$. For every $c > 0$ let

$$X_c = \{ x : a(s_i \# x \# y_x) \geq a(s_i)/c \}$$

Claim 4: $|X_c| \leq c$
Fix $b,d > 0$. Choose $l$ random linear functions $H=\{h_1, \ldots, h_l\}$, $h_i: \Sigma^m \rightarrow \Sigma^b$ and $l^2$ random strings $Z \subseteq \Sigma^b$. Pick any $x \in H^{-1}(Z)$ and any $y \in \Sigma^m$. Let $s_{i+1} = s_i \# x \# y$.

We now describe a collection of events corresponding to exceptional luck on Merlin’s part.

Call the following event $E_{i+1}$:

$$a(s_{i+1}) \geq \frac{a(s_i)}{2^b/d}$$
Claim 5: \( \Pr[E_i] \leq \frac{l^3}{d} \)

Proof: Let \( c = \lfloor d/2^b \rfloor \). Then \( |X_c| \leq 2^b / d \) by claim 4. Since \( a(s_i \# x \# y_x) \geq a(s_{i+1}) \) by the definition of \( y_x \), if \( a(s_{i+1}) \geq a(s_i)/(2^b/d) \) then \( x \in X_c \). Since \( x \in H^{-1}(Z) \),

\[
Pr \left[ a(s_{i+1}) \geq \frac{a(s_i)}{2^b/d} \right]
\]

\[
= \Pr[x \in X_c \cap H^{-1}(Z)]
\]

\[
= \Pr[H(X_c) \cap Z \neq \emptyset]
\]

\[
\leq \frac{l^3}{d}
\]

by the approximate lower bound lemma part 2b. \( \blacksquare \)

\textbf{if} \( |C| \leq 2^b/d, \ d>0 \), \textbf{then} \( \Pr[H(C) \cap Z \neq \emptyset] \leq \frac{l^3}{d} \)  

(Claim 5/6) (Full proof \( w \not\in L \))
Fix $s_g$. Choose $l$ random linear functions $H = \{h_1, \ldots, h_l\}$, $h_i : \Sigma^l \rightarrow \Sigma^{b_g+1}$ and $l^2$ random strings $Z \subseteq \Sigma^{b_g+1}$. Pick any $r \in H^{-1}(Z)$. Call the following event $E_{g+1}$:

$$2^l \alpha(s_g) \leq 2^{b/d} \text{ and } (V*P)(w,r) \text{ accepts via } s_g$$

**Claim 6:**

$$\Pr[E_{g+1}] \leq l^3/d$$

**Proof:** By the approximate lower bound lemma part 2b, since $|\{r: (V*P)(w,r) \text{ accepts via } s_g\}| = 2^l \alpha(s_g)$. □

**if** $|C| \leq 2^{b/d}$, $d > 0$, **then** $\Pr[H(C) \cap Z \neq \emptyset] \leq l^3/d$  

( Claim 6/6 )

( Full proof $w \notin L$ )
In any run of $A$ and $M$, event $E_i$ may occur during round $i$, where $b=b_i$ for $i \leq g+1$. The probability that each occurs is at most $l^3/d$ and therefore the probability that any occurs is at most $(g+1)l^3/d$. Choose $d=3(g+1)l^3$.

Then $\Pr[\exists iE_i \text{ occurs}] \leq 1/3$.  

(Full proof $w \not\in L$)
Assume no $E_i$ occurs. Then we show that $A$ will reject, provided that $\Pr[V(w) \text{ accepts}] \leq e$.

Since $\forall i \leq g, \neg E_i$, we have:

$$\frac{a(s_0)}{\prod_{i \leq g} (2^{b_i}/d)} \geq a(s_g)$$

Since $\neg E_{g+1}$:

$$(V*P)(w,r) \neq \text{ accept}$$

or

$$2^l a(s_g) \geq 2^{b_{g+1}}/d$$
Thus if \((V \ast P)(w,r)\) accepts, combining the above:

\[2^l a(s_0) \geq \prod_{1 \leq i \leq g+1} (2^{b_i} / d)\]

so, since \(l \geq g + 1\), taking logs:

\[l + \log a(s_0) \geq \sum b_i - (g + 1) \log d \]

\[\geq \sum b_i - (g + 1)\]

\[\geq \sum b_i - 10g \log l\]

but

\[a(s_0) = \Pr[V(w) \text{ accepts}] \leq e \leq l^{-12g}\]

so

\[l - 12g^2 \log l \geq \sum b_i - 10g \log l\]
Thus
\[ \sum b_i \leq l - 2g \log l < l - g \log l \]

Recall that Arthur only accepts if \((V*P)(w,r)\) accepts and \(\sum b_i \geq l - g \log l\). Therefore if \(\forall \ i \leq g+1, \ E_i\) occurs and \(\Pr[V(w)\text{ accepts}] \leq e\), then Arthur will reject. Hence \(\Pr[A(w)\text{ accepts}] \leq 1/3\). \(\blacksquare\)