1. （25 分） 设 $G(s)$ 是一个安全的 PRG，输出空间为 $\{0,1\}^n$ 。
定义另一个 PRG 为 $G_1(s) := (G(s), G(s))$ ，输出空间为 $\{0,1\}^{2n}$ 。
请问 $G_1(s)$ 是安全的 PRG 吗？
请**证明**你的结论

证明：
定义如下两个实验：
EXP(0)：$r= G_1(s) := (G(s), G(s))$
EXP(1)： $r\xleftarrow{R}\{0,1\}^{2n}$
挑战者将 r 传递给攻击者作为输入。
构造攻击者算法 A，A 执行以下步骤：
① $r_1||r_2\leftarrow r$，$r_1\in\{0,1\}^n$ ，$r_2\in\{0,1\}^n$
② $x\leftarrow r_1\oplus r_2$
③ 若 x 为 0，返回 0，表示 r_1 和 r_2 为 G(s);否则返回 1，表示 r_1 和 r_2 为真随机序列

$$Adv_{PRG}[A,G_1] = |\Pr[EXP(0)=1] - \Pr[EXP(1)=1]| = |0-(1-\frac{1}{2^n})| = 1-\frac{1}{2^n}$$

由于优势不可忽略，因此，$G_1(s)$不是安全的 PRG

1+.补充. （25 分） 设 $G(s)$ 是一个安全的 PRG，输出空间为 $\{0,1\}^n$ 。
定义另一个 PRG 为 $G_2(s_1,s_2) := (G(s_1), G(s_2))$。
请问 $G_2(s)$是安全的 PRG 吗？
请**证明**你的结论

证明：

定义如下三个实验：

EXP(0)：　$r=G_2(s):=(G(s_1),G(s_2))$

EXP(0.1)：$r=(r_1,G(s_2))$　　$r_1\xleftarrow{R}\{0,1\}^n$

EXP(1)：　$r=(r_1,r_2)$　　r_1、$r_2\xleftarrow{R}\{0,1\}^n$

(1) 假设存在算法 A_1 能区分 EXP(0)和 EXP(0.1)，其中 A_1 的参数为 r，$r\in\{0,1\}^{2n}$：若 $A_1(r)=0$，则表示 A_1 的参数 r 来自 EXP(0)中的串；若 $A_1(r)=1$，则表示 A_1 的参数 r 来自 EXP(0.1)中的串。

我们可以通过算法 A_1 来构造一个算法 B_1，算法 B_1 可以用来区分 G(s)和真随机序列，给定 r_1' 作为 B_1 的输入，其中 $r_1'\in\{0,1\}^n$，然后定义算法 B_1：

① 令 $r_2'=G(s_2)$，$s_2\xleftarrow{R}K$，K 为种子空间，$r_2'\in\{0,1\}^n$

② Call Alg.$A_1(r_1',r_2')$

③ 若 $A_1(r_1',r_2')=0$，则返回 0，表示 r_1' 为 G(s)；否则返回 1，表示 r_1' 为真随机序列

$\boxed{Adv_{PRG}[G,\ B_1]=AdvA_1=|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]|>\varepsilon_1(\varepsilon_1\text{不可忽略})}$

因此 G(s)为一个不安全的 PRG，与已知相矛盾，假设不成立，所以 EXP(0)和 EXP(0.1)不可区分，所以 $|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]|\le\varepsilon_1$（$\varepsilon_1$ 可忽略）

(2) 假设存在算法 A_2 能区分 EXP(0.1)和 EXP(1)，其中 A_2 的参数为 r，$r\in\{0,1\}^{2n}$：若 $A_2(r)=0$，则表示 A_2 的参数 r 来自 EXP(0.1)中的串；若 $A_2(r)=1$，则表示 A_2 的参数 r 来自 EXP(1)中的串。

我们可以通过算法 A_2 来构造一个算法 B_2，算法 B_2 可以用来区分 G(s)和真随机序列，给定 r_2' 作为 B_2 的输入，其中 $r_2'\in\{0,1\}^n$，然后定义算法 B_2：

① 取 $r_1'\xleftarrow{R}\{0,1\}^n$

② Call Alg.$A_2(r_1',r_2')$

③ 若 $A_2(r_1',r_2')=0$，则返回 0，表示 r_2' 为 G(s)；否则返回 1，表示 r_2' 为真随机序列

$\boxed{Adv_{PRG}[G,B_2]=AdvA_2=|Pr[EXP(0.1)=1]-Pr[EXP(1)=1]|>\varepsilon_2(\varepsilon_2\text{不可忽略})}$

因此 G(s)为一个不安全的 PRG，与已知相矛盾，假设不成立，所以 EXP(0.1)和 EXP(1)不可区分，所以 $|Pr[EXP(0.1)=1]-Pr[EXP(1)=1]|\le\varepsilon_2$（$\varepsilon_2$ 可忽略）

(3) 定义算法 A 用来区分 EXP(0)和 EXP(1)，根据(1)和(2)中的优势可以得出：

$$
\begin{aligned}
Adv_{PRG}[G_2,A]&=|Pr[EXP(0)=1]-Pr[EXP(1)=1]|\\
&=|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]+Pr[EXP(0.1)=1]-Pr[EXP(1)=1]|\\
&\leqslant|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]|+|Pr[EXP(0.1)=1]-Pr[EXP(1)=1]|\\
&\leqslant\varepsilon_1+\varepsilon_2\ (\varepsilon_1+\varepsilon_2\text{可忽略})\\
&=2Adv_{PRG}[G,B]
\end{aligned}
$$

所以 EXP(0)和 EXP(1)不可区分，因此 $G_2(s)$是一个安全的 PRG。

2. （25 分） 设 G(s) 是一个安全的 PRG，输出空间为 $\{0,1\}^n$ 。
定义另一个 PRG 为 $G_2(s_1||s_2)$:= $(s_1, G(s_2))$ 。
请问 $G_2(s)$是安全的 PRG 吗？
请**证明**你的结论

证明：

定义如下两个实验：

EXP(0)： $r= G_2(s_1||s_2):= (s_1, G(s_2))$，$s_1, s_2 \xleftarrow{R} K$，K 为种子空间

EXP(1)： $r \xleftarrow{R} \{0,1\}^{n+|s1|}$

假设存在算法 A 能区分 EXP(0)和 EXP(1)，挑战者将 r 传递给攻击者作为输入，若 A(r)=0，则表示 A 的参数 r 来自 EXP(0)；若 A(r)=1，则表示 A 的参数 r 来自 EXP(1),是真随机序列。

我们可以通过算法 A 来构造一个算法 B，算法 B 可以用来区分 G(s)和真随机序列，给定 r_1' 作为 B 的输入，其中 $r_1' \in \{0,1\}^n$，然后执行以下步骤：

(1)Call Alg. A(s_1'，r_1')，其中 $s_1' \xleftarrow{R} K$，K 为种子空间

(2)若 A(s_1'，r_1')=0，则返回 0，表示 r_1' 来自 G(s)；否则返回 1，表示 r_1' 为真随机序列

$\boxed{Adv_{PRG}[G,B]= Adv_{PRG}[G_2,A]=|Pr[EXP(0)=1]-Pr[EXP(1)=1]|> \varepsilon \text{ (} \varepsilon \text{ 不可忽略)}}$

因此 G(s)为一个不安全的 PRG，与已知相矛盾，假设不成立，所以 $G_2(s)$和真随机序列不可区分，因此 $G_2(s)$是一个安全的 PRG。

3. （25 分） 设 G(s) 是一个安全的 PRG，输出空间为 $\{0,1\}^n$ 。
定义另一个 PRG 为 $G_3(s)$:= G(s) ⊕ 1^n 。
请问 $G_3(s)$是安全的 PRG 吗？
请**证明**你的结论

定义如下两个实验：

EXP(0)： r= $G_3(s) := G(s) \oplus 1^n$

EXP(1)： $r \xleftarrow{R} \{0,1\}^n$

假设存在算法 A 能够区分 EXP(0)和 EXP(1)，若 A(r)=0，则表示 A 的参数 r 是来自 EXP(0)中的串；若 A(r)=1，则表示 A 的参数 r 是来自 EXP(1)中的串；我们可以通过 A 来定义算法 B，B 是用来区分 G(S)和真随机序列，给定 r_1 为 B 的输入，其中 $r_1 \in \{0,1\}^n$，然后执行如下步骤：

① Call Alg. A($r_1 \oplus 1^n$)

② 若 A($r_1 \oplus 1^n$)=0，则返回 0，表示 r_1是 G(s)；否则返回 1，表示 r_1是真随机序列

注意：若r_1=G(s)，则 $r_1 \oplus 1^n$ 是 $G_3(s)$，若$r_1 \xleftarrow{R} \{0,1\}^n$，则 $r_1 \oplus 1^n$ 依旧是真随机序列。

所以：$Adv_{PRG}[G, B] = Adv_{PRG}[G_3, A] = |\Pr[EXP(0)=1] - \Pr[EXP(1)=1]| > \varepsilon$($\varepsilon$不可忽略)

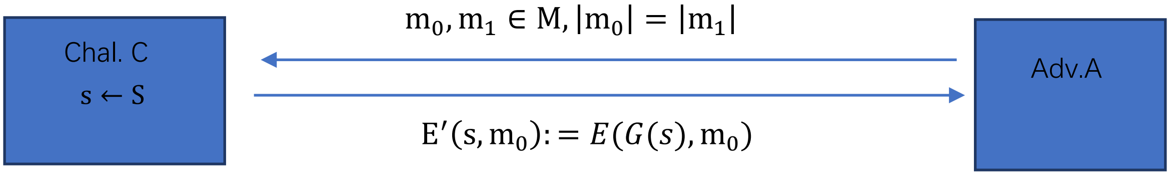
因此 G 是一个不安全的 PRG，与已知相矛盾，因此假设不成立，所以 EXP(0)和 EXP(1)不可区分，所以 $G_3(s)$是一个安全的 PRG。

4. （25 分） 设 G ： S → R 是安全的 PRG.
设 （E, D） 是语义安全的对称加密方案, E: K × M → C.
假设 K = R.
构造一个新的对称加密方案 （E’ , D’ ）, E’ : S × M → C, 其中
E’ （s, m） := E(G(s), m) , D’ （s, c） := D(G(s), c).
请证明 E’ 也是语义安全的

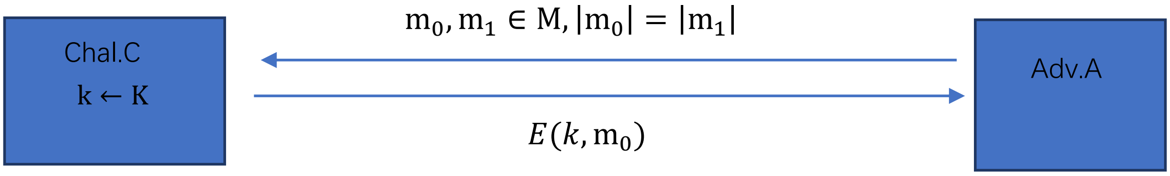
证明：

定义如下四个实验：

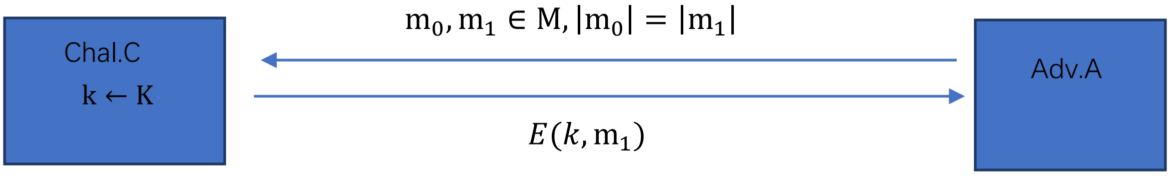
EXP（0）：



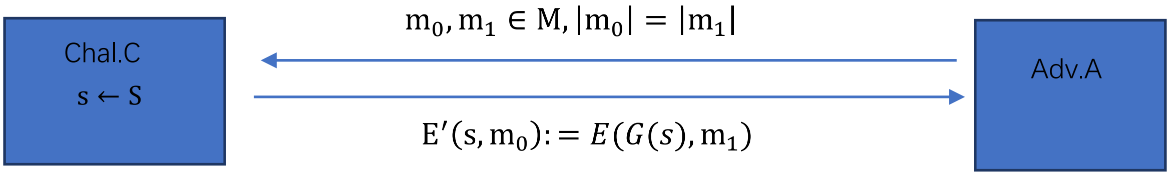
EXP（0.1）：



EXP（1.1）：



EXP（1）：



(1)假设存在算法 A_1能区分 EXP(0)和 EXP(0.1)，其中 A_1的参数为 r，r= $E(G(s),m_0)$或者 r=$E(k,m_0)$
若 $A_1(r)=0$，则表示 A_1的参数 r= $E(G(s),m_0)$；若 $A_1(r)=1$，则表示 A_1的参数 r=$E(k,m_0)$。

我们可以通过算法 A_1来构造一个算法 B_1，算法 B_1可以用来区分 G(s)和真随机序列，给定 r_1作为 B_1的输入，其中 $r_1 \in \{0,1\}^n$，A_1和 B_1进行如下交互：

① A_1 把$m_0, m_1 \in M, |m_0| = |m_1|$发送给 B_1

② B_1选择一个消息m_0, 连同自己的输入r_1作为 E 的输入执行对称加密方案 E

③ B_1将 $E(r_1, m_0)$发送给 A_1

④ 若 A_1返回 0, 则 B_1返回 0，表示 r_1为 G(s)；否则返回 1，表示 r_1为真随机序列

所以：

$Adv_{PRG}[G, B_1] = Adv A_1 = |Pr[EXP(0)=1]-Pr[EXP(0.1)=1]| > \varepsilon_1$（$\varepsilon_1$不可忽略）

因此 G(s)为一个不安全的 PRG，与已知相矛盾，假设不成立，所以 EXP(0)和 EXP(0.1)不可区分，所以|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]| ≤ ε_1（ε_1可忽略）= Adv_{PRG}[G, B]

(2）因为（E,D）是一个语义安全的对称加密方案，

所以:|Pr[EXP(0.1)=1]-Pr[EXP(1.1)=1]| ≤ ε_2（ε_2可忽略）=Adv_{SS}[E,A]

(3) 假设存在算法A_2能区分EXP(1.1)和EXP(1)，其中A_2的参数为r，$r = E(k, m_1)$或者$r = E(G(s), m_1)$：若 A_2(r)=0，则表示 A_2的参数$r = E(G(s), m_1)$；若 A_2(r)=1，则表示 A_2的参数$r = E(k, m_1)$。

我们可以通过算法 A_2来构造一个算法 B_2，算法 B_2可以用来区分 G(s)和真随机序列，给定 r_2作为 B_2的输入，其中 $r_2 \in \{0,1\}^n$，然后 A_2和 B_2进行如下交互：

① A_2 把$m_0, m_1 \in M, |m_0| = |m_1|$发送给 B_2

② B_2选择一个消息 m_1 ，连同自己的输入r_2作为 E 的输入执行对称加密方案 E

③ B_2将 $E(r_2, m_1)$发送给 A_2

④ 若 A_2返回 0, 则 B_2返回 0，表示 r_2为 G(s)；否则返回 1，表示 r_2为真随机序列.

所以：

Adv_{PRG}[G, B_2]= $Adv$$A_2$=|Pr[EXP(1.1)=1]-Pr[EXP(1)=1]|> ε_3(ε_3不可忽略)

因此 G(s)为一个不安全的 PRG，与已知相矛盾，假设不成立，所以 EXP(0)和 EXP(0.1)不可区分，所以|Pr[EXP(1.1)=1]-Pr[EXP(1)=1]| ≤ ε_3（ε_3可忽略）= Adv_{PRG}[G, B]

(4) 定义算法 A’用来区分 EXP(0)和 EXP(1)，根据(1)、(2)、（3）中的优势可以得出：

Adv_{SS}[E’, A’]=|Pr[EXP(0)=1]-Pr[EXP(1)=1]|

=|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]+Pr[EXP(0.1)=1]-Pr[EXP(1.1)=1]

+Pr[EXP(1.1)=1]-Pr[EXP(1)=1]|

≤|Pr[EXP(0)=1]-Pr[EXP(0.1)=1]|+|Pr[EXP(0.1)=1]-Pr[EXP(1.1)=1]|

+| Pr[EXP(1.1)=1]-Pr[EXP(1)=1]|

≤ ε_1+ ε_2+ ε_3(ε_1+ ε_2+ ε_3可忽略)

=2 Adv_{PRG}[G, B]+ Adv_{SS}[E,A]

所以 EXP(0)和 EXP(1)不可区分，因此 E’是语义安全的。