MTAT.07.003 Cryptology II
Spring 2012 / Exercise session ?? / Example Solution

Exercise (Simulator for the QNR-ZKD protocol). *Let n be a composite number with a factorisation $n = pq$ known to the prover $\mathcal{P}$. Let $v \in \mathbb{Z}_n^*$ be a number for which the prover wants to prove that it is quadratic non-residue. Let $(\mathsf{Gen}, \mathsf{Com}, \mathsf{Open})$ be a perfectly binding and computationally hiding commitment. Show that the the following zero-knowledge protocol*

$$
\begin{array}{lcl}
\mathcal{P} & & \mathcal{V} \\
\mathsf{pk} \leftarrow \mathsf{Gen} & & b \xleftarrow{u} \{0,1\}\,, \\
 & \xleftarrow{\qquad c = r^2 v^b \qquad} & r \xleftarrow{u} \mathbb{Z}_n^* \\
\overline{b} \leftarrow \mathsf{IsQnr}_{p,q}(c) & & \\
(\overline{c}, \overline{d}) \leftarrow \mathsf{Com}(\overline{b}) & \xrightarrow{\qquad \mathsf{pk}, \overline{c} \qquad} & \\
 & \xleftarrow{\qquad r, b \qquad} & \\
c \stackrel{?}{=} r^2 v^b & & \\
 & \xrightarrow{\qquad \overline{d} \qquad} & \mathsf{Open}_{\mathsf{pk}}(\overline{c}, \overline{d}) \stackrel{?}{=} b
\end{array}
$$

where the verifier $\mathcal{V}$ releases r, b is simulatable. For that construct first a semi-efficient algorithm $\mathcal{K}^{\mathcal{V}_*}$ for extracting r, b that correspond to initial message c. Next show that the following simulator construction

$$
\begin{array}{l}
\mathcal{V}_\circ(\phi) \\
\left[\begin{array}{l}
\omega \leftarrow \Omega \\
c \leftarrow \mathcal{V}_*(\phi; \omega) \\
(b_*, r_*) \leftarrow \mathcal{K}^{\mathcal{V}_*}(\phi; \omega) \\
\alpha \leftarrow \mathcal{V}_* \\
\beta \xleftarrow{u} \mathcal{B} \\
\gamma \leftarrow \mathcal{V}_*(\beta) \\
\text{if } \mathsf{Ver}(\alpha, \beta, \gamma) = 0 \text{ then } \textbf{\textit{return}}\ \mathcal{V}_*(\bot) \\
\text{if } \beta_* \neq \bot \text{ then } \textbf{\textit{return}}\ \mathcal{V}_*(b_*) \\
\text{else } \textbf{\textit{return}}\ \bot
\end{array}\right.
\end{array}
$$

can create an output distribution $\psi_\circ$ that is computationally $(t_\circ, \varepsilon_\circ)$-distant from the output distribution of malicious verifier $\mathcal{V}_*$ that interacts with the honest prover. Also, estimate how the running-time of the simulator depends on the desired distance $\varepsilon_\circ$.

Solution.