Exercise 1:

- write $n_i = \lfloor n_i/2 \rfloor \cdot 2 + (n_i \bmod 2)$

$$\prod_{i \in [t]} g_i^{n_i} = \Big(\prod_{i \in [t]} g_i^{\lfloor n_i/2 \rfloor} \Big)^2 \cdot \underbrace{\Big(\prod_{i \in [t]} g_i^{\overbrace{(n_i \bmod 2)}^{\in \{0,1\}}} \Big)}_{\text{max. } 2^t \text{ values}}$$

- precompute $V_b = \prod_{i \in [t]} g_i^{b_i}$ for $b = (b_i)_i \in \{0,1\}^t$
- store values in table $T[b] = V_b$
- compute recursively $\prod_{i \in [t]} g_i^{\lfloor n_i/2 \rfloor} =: A$ using table T
- output $A^2 \cdot T[b]$ with $b = (n_i \bmod 2)_i$

Exercise 2:

1. decryption of $c = (c_1, c_2)$: $m = c_1 \cdot c_2^{-sk}$
2. easy dlog: $\mathrm{dlog}_g(pk) = sk$ and decrypt with sk
   hard dlog: unknown (cf. next question)
3. we perform a reduction
   - suppose there is some $\mathcal{A}$, such that:
     - $m \xleftarrow{u} \mathbb{Z}_p$
     - $(pk, sk) \leftarrow \mathrm{KeyGen}(1^\lambda)$
     - $c = \mathrm{Enc}(pk, m; r)$
     - $m \leftarrow \mathcal{A}(g, pk, c)$
   - receive (g, X, Y), $X = g^x, Y = g^y$
   - set $z \leftarrow \mathbb{Z}_p$
   - set $c = (g^z, X)$
   - $m \longleftarrow \mathcal{A}(g, X, c)$
   - if $\mathcal{A}$ was correct:
     $m = g^z / X^y \Rightarrow X^y = g^z/m$
   - output g^z/m
   - this algorithm breaks the hardness assumption
     $\Rightarrow$ such $\mathcal{A}$ can't exist

4. $\mathrm{Enc}(pk, m) \cdot \mathrm{Enc}(pk, m') = (m \cdot m' \cdot pk^{r+r'}, g^{r+r'})$

5. perform reduction:
- suppose there is some $\mathcal{A}$, such that:
  - $b \xleftarrow{u} \{0,1\}$,
  - $(pk, sk) \longleftarrow KeyGen(1^\lambda)$
  - $c_b = Enc(pk, m_b)$
  - $b' \leftarrow \mathcal{A}(pk, c_b)$ with $b = b'$ with prob. ε
- obtain (g, X, Y, C) for $C \in \{g^{xy}, g^c\}$
- draw $b \xleftarrow{u} \{0,1\}$ - suppose there is some $\mathcal{A}$. such that:
- set $pk = X$, $c_b = (m_b \cdot C, Y)$
- output 1 iff $b = b' \longleftarrow \mathcal{A}(pk, c_b)$
- note: if $C = g^c$, then m_b is information theoretically hidden
  $\hookrightarrow Pr[b' = b \mid C = g^c] = \frac{1}{2}$
  if $C = g^{xy}$, then $b = b'$ with prob. ε
  $\hookrightarrow Pr[b' = b \mid C = g^{xy}] = \varepsilon$
- thus: $Pr[\text{our guess is correct}] = Pr[b' = b \mid C = g^c] \cdot \frac{1}{2}$
  $+ Pr[b' = b \mid C = g^{xy}] \cdot \frac{1}{2}$
  $= \frac{1}{4} + \varepsilon/2$
  $\Rightarrow \frac{\varepsilon}{2} \approx \frac{1}{4}$ under assumption

6. m is uniquely determined by (pk, c)
7. cachant: no, because we can compute sk,
   then decrypt to m' and check $m' = m_0$ or $m' = m_1$
   binding: yes (see above)
8. - $c = (g^m \cdot pk^r, g^r)$
   - can only decrypt for small m
9. - $c = (g^m \cdot pk^r)$
   - contraignant: Dlog
     - given m, m', r, r' with $m \neq m'$: $g^m \cdot pk^r = g^{m'} \cdot pk^{r'}$
     - have $g^{m-m'} = pk^{r'-r} \Rightarrow r \neq r'$ and $g^x = pk$ for $x = m-m'/_{r'-r}$
   - hiding: unconditional
     - pk^r functions as a one-time pad of g^m $(c \sim U_G)$
10. Setup: $n = p \cdot q$, $e \cdot d = 1 \mod \varphi(n)$, $pk = e$, $sk = d$
    Encrypt: $c = m^e \mod n$
    Decrypt: $m = c^d \mod n$
    - cachant: unsure (assumption is called RSA-assumption)
    - binding: yes

Exercise 3:

A) brute force (try all possible $x \in \mathbb{Z}_p$, check $g^x = h$)

B) - $q \in (0, m)$
- $r \in [0, m]$
- precompute $hg^{-m \cdot q}$ for all $q \in (0, m)$, save in table T
- test if $g^r \in T$ for all $r \in [0, m]$, if hit:

$$g^r = h \cdot g^{-m \cdot q} \Rightarrow g^{r + m \cdot q} = h \Rightarrow r + m \cdot q = x$$

C)

1. show: $\alpha_i = g^{x_i} h^i$ $\quad \beta_i = g^{y_i} h^{2i}$

$x_i = \sum_{j=1}^{i-1} F(\alpha_j)$, $\quad y_i = \sum_{j=1}^{2i-1} F(\alpha_j)$ $\quad (\Rightarrow \alpha_{2i} = \beta_i,\ x_{2i} = y_i)$

before while: $\alpha_1 = h = g^{x_1} \cdot h$ as $x_1 = 0$

$$\beta_1 = H(h) = h^2 \cdot g^{F(\alpha)} = h^2 \cdot g^y$$
$$x_1 = 0$$
$$y_1 = F(\alpha_1)$$

after iteration: $\alpha_{i+1} = H(\alpha_i) = \alpha_i \cdot h \cdot g^{F(\alpha_i)}$

$$= g^{x_i + F(\alpha_i)} \cdot h^{i+1}$$
$$= g^{x_{i+1}} \cdot h^{i+1} \quad \text{with } x_{i+1} = \sum_{j=1}^{i} F(\alpha_i)$$

then also:

$$\beta_{i+1} = H(H(\beta_i))$$
$$= H(\beta_i h \cdot g^{F(\beta_i)})$$
$$= H(\underbrace{g^{y_i} \cdot h^{2i+1} \cdot g^{F(\beta_i)}}_{\alpha_{2i+1}})$$

(*)

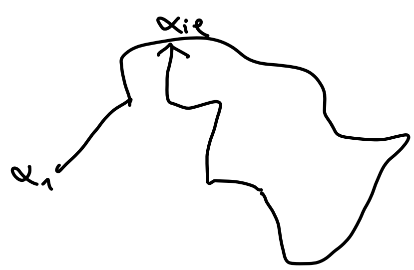
$$= \alpha_{2i+1} \cdot h^{2(i+1)} \cdot g^{F(\alpha_{2i+1})}$$
$$= g^{x_{2i+1}} \cdot h^{2(i+1)} \cdot g^{F(\alpha_{2i+1})}$$
$$= g^{x_{2(i+1)}} \cdot h^{2(i+1)}$$
$$= g^{y_{i+1}} \cdot h^{2(i+1)}$$

(*) $\beta_i = \alpha_{2i}$, $y_i = x_{2i}$

2. $\alpha_i = \beta_i \Rightarrow \alpha_i = g^{x_i} h^i = g^{y_i} h^{2i} = \beta_i$

$\Rightarrow g^{x_i - y_i / i} = h$ as $i \neq 0$

3. - there are p possible values for $\alpha_i \Rightarrow \exists i, j \in (1, p+1): \alpha_i = \alpha_j$
- let $\ell = j - k$
- note that $\alpha_{\ell \cdot i} = \alpha_{2\ell \cdot i}$ for all $i \in \mathbb{N}$
- there are ℓ values between $\alpha_k, \dots, \alpha_{j-1}$
- statement follows (as one must be multiple of ℓ)



4. - α_i defines random walk in G
- first collision after $O(\sqrt{p})$ in expectation (birthday paradox)
- that is, $\mathbb{E}[j] = O(\sqrt{p})$

$\Rightarrow \ell = O(\sqrt{p}) \overset{3}{\Rightarrow}$ statement

Exercise 4:

- $h \cdot h'$ is a random element in G
- with prob. $|E|/|G| \geq \varepsilon$ we have $h \cdot h' \in E$
- in that case: $w = \mathrm{dlog}_g(h \cdot h')$

$\Rightarrow h \cdot h' = g^w$

$\Rightarrow h = g^{w-c}$