1. (a) Assume we have an oracle for CLIQUE(G,k). Let $\omega = \omega(G)$ be the size of the maximal clique in G; note that $1 \leq \omega \leq n$.

Algorithm 1 FIND-MAX-CLIQUE(G)

$\omega \leftarrow \omega(G)$ {use any search algorithm (naive, linear, binary)}
$K \leftarrow V$
for $v \in K$ **do**
  if CLIQUE$(G[K-\{v\}],\omega) = true$ **then**
    $K \leftarrow K - \{v\}$
return K

Correctness:
The algorithm erases all vertices not crucial so the survival of the largest clique, so at the end we have $|K| = \omega(K) = \omega(G)$ and K is the requested clique.

Time Analysis:
The oracle is queried $\leq 2n$ times.

(b) The interesting case is of course $G_1 \sim G_2$. Denote G_1's vertices by $\{u_1, u_2, \ldots, u_n\}$ and G_2's vertices by $\{v_1, v_2, \ldots, v_n\}$, with the isomorphism being $v_i \sim u_{\pi(i)}$.
We need a set of n distinguishable gadgets $\{H_i\}_{i=1}^n$ that haven't appeared in G_1, G_2, for instance $H_i = K_{n+i}$.

Algorithm 2 FIND-ISOMORPHISM(G_1, G_2)

for $i \leftarrow 1$ to n **do**
  connect H_i to u_i in G_1
  for $k \leftarrow 1$ to n **do**
    $G_2' \leftarrow G_2$
    connect H_i to v_k in G_2'
    if ISOMORPHIC(G_1, G_2') **then**
      $\pi(i) \leftarrow k$
      $G_2 \leftarrow G_2'$
      break from the k for-loop
  if we got here not by breaking from the loop (can only happen for $k = 1$), return $false$
return π

Correctness:
In step i we recover $\pi(i)$ assuming we know $\pi(j)$ for $j < i$ by checking all possibilities. Any isomorphism $G_1 \overset{\tau}{\sim} G_2$ must have $\tau(j) = \pi(j)$ for $j < i$ due to the gadget H_j. By the end of step n, we know π entirely.

Time Analysis:
The oracle is queried $\leq n^2$ times.

Note that by erasing vertices (instead of marking them) we might get the wrong permutation (e.g., consider a path of length 3).

2. We show that a c-approximation algorithm B can be used to solve $\text{GAP}_{[\alpha,\beta]}$-A, so B cannot be polynomial unless $P = NP$.

   Algorithm:
   Given an instance x of $\text{GAP}_{[\alpha,\beta]}$-A, we return $true$ if and only if $B(x) < \beta$.

   Correctness:
   If $x \in YES$, then there is a solution for x of size $< \alpha$, hence $B(x) \leq cOPT(x) < c\alpha \leq \beta$ and we return $true$;

   If $x \in NO$, then any solution for x, including $B(x)$, is of size $\geq \beta$, hence we return $false$.

3. We know that E3-SAT is NP-hard to approximate within any $c > \frac{7}{8}$;
   However, $\text{GAP}_{[\epsilon,\frac{1}{2}]}$-E3-SAT $\in P$ since $NO = \emptyset$ as $\geq \frac{7}{8}$ of the clauses can be always satisfied.
   Now select ϵ small enough such that $2\epsilon < \frac{7}{8} < c$.

4. Bounded-degree $\text{GAP}_{[\alpha,\beta]}$-MAX-IS is known to be NP-hard for some $0 < \alpha < \beta < 1$. Since a vertex cover is always the complement of an independet set, MIN-VC is the complement of a MAX-IS. We conclude that bounded-degree $\text{GAP}_{[1-\beta,1-\alpha]}$-MIN-VC is NP-hard as well (use the trivial reduction) hence bounded-degree MAX-VC is NP-hard to approximate within $c = \frac{1-\alpha}{1-\beta} (> 1)$.