Smart Contract State: $c_1, c_2, \lambda_1, \lambda_2 \in \mathbb{Z}$

1) $\mathsf{Constructor}(\lambda_1, \lambda_2)$:
   Store input values λ_1, λ_2 to the corresponding state variables
   $c_1 \leftarrow \mathsf{InitAcc}(\lambda_1)$
   $c_2 \leftarrow \mathsf{InitAcc}(\lambda_2)$
2) $\mathsf{Register}(id, pk, W_2, c_{add_1}, W_{add_1}, c_{add_2}, W_{add_2})$:
   if $\mathsf{sizeof}(id) \neq \lambda_2 \vee \mathsf{CheckUpdate}(c_2, c_{add_2}, W_{add_2}, id) = 0 \vee \mathsf{sizeof}(id, pk) \neq \lambda_1 \vee \mathsf{CheckUpdate}(c_1, c_{add_1}, W_{add_1}, (id, pk)) = 0 \vee \mathsf{VerifyNonMem}(c_2, W_2, id) = 0$
      return fail
   endif
   $c_1 \leftarrow c_{add_1}$
   $c_2 \leftarrow c_{add_2}$
3) $\mathsf{Revoke}(id, pk, W_1, \sigma_{sk}(pk), c_{del_1}, W_{del_1}, c_{del_2}, W_{del_2})$:
   if $\mathsf{sizeof}(id) \neq \lambda_2 \vee \mathsf{sizeof}(id, pk) \neq \lambda_1 \vee \mathsf{VerifyMem}(c_1, W_1, (id, pk)) = 0 \vee \mathsf{VerifySig}(\sigma_{sk}(pk), pk) = 0 \vee \mathsf{CheckUpdate}(c_1, c_{del_1}, W_{del_1}, (id, pk)) = 0 \vee \mathsf{CheckUpdate}(c_2, c_{del_2}, W_{del_2}, id) = 0$
      return fail
   endif
   $c_1 \leftarrow c_{del_1}$
   $c_2 \leftarrow c_{del_2}$
4) $\mathsf{RetrieveState}()$:
   return $(c_1, c_2, \lambda_1, \lambda_2)$