Algorithm 1 Dynamic Programming State Transition

for each_day **in** [initial_time, end_time] **do**
    is_gold_market_open_today ← check_if_market_open(gold_date, current_day)
    end_state[current_day] ← [0.0, 0.0, 0.0]
    next_type[current_day + 1] ← [0, 0, 0]
    if is_gold_market_open_today **then**
        compute_max_profit_and_optimal_type
    else
        compute_max_profit_and_optimal_type ▷ Gold investment profit is −1
    end if ▷ update_state
    today_gold_price, next_gold_price, today_bitcoin_price, next_bitcoin_price ← get_prices(current_day)
    end_state[current_day + 1] ← [computed_profit, . . .]
end for