

```

// Declare variables
DECLARE feeding_times[ ] AS ARRAY OF TIME
DECLARE current_time AS TIME
DECLARE food_level AS INTEGER // serving
DECLARE bowl_weight AS INTEGER // grams
DECLARE empty_bowl_weight AS INTEGER ← calibrated empty bowl weight


//Step 1: Start the program
BEGIN

//Step 2: Get feeding schedule
INPUT feeding_times[ ] from staff //list of scheduled feeding times for the day (24h
format)

//Step 3: Main loop
WHILE NOT all feeding_times completed DO
    //Step 4: Get current time
    READ current_time from real_time_clock

    //Step 5: Check if it is feeding time
    IF current_time matches any feeding_time in feeding_times THEN
        //Step 6: Check food storage
        READ food_level from the sensor

        //Step 7: If food is low
        IF food_level <= 1 serving THEN
            //Step 8: Alert staff
            send_alert("Food level low in storage")
        END IF

        //Step 9: Dispense food
        DISPENSE_FOOD()
    END IF
END WHILE

```

//Step 10: Wait 15 minutes

WAIT(15 minutes)

//Step 11: Read bowl weight

READ from bowl_weight from the sensor

//Step 12: Check if food is left in bowl

WHILE bowl_weight > empty_bowl_weight DO

 //Step 13: Alert staff

 send_alert("Food remains in bowl")

 //Step 14: Wait 1 minute before checking again

 WAIT(1 minute)

 //Step 11 (repeat): Read bowl weight again

 READ bowl_weight from the sensor in the bowl

END WHILE

END IF

//Step 15: Check if all feeding times are done

IF all feeding_times completed THEN

 EXIT_LOOP

END IF

END WHILE

//Step 16: End the program

END