

Pseudocode FOR Task1, 2 & 3

Task 1

Allow auction company to enter item details.

```
1  product_selling <-- 0
2  OUTPUT("Welcome to auction software.")
3
4  OUTPUT "Enter how many items you want to sell?"
5  INPUT product_selling
6  WHILE product_selling < 10 DO product_selling
7      OUTPUT "Enter how many items you want to sell?"
8      INPUT product_selling
9      OUTPUT "Error, please enter more than 10 products."
10 ENDWHILE
11 OUTPUT "Number valid."
12
13 DECLARE name_list : ARRAY[1] OF STRING
14 DECLARE bid_number_list : ARRAY[1 : product_selling] OF INTEGER
15 DECLARE description_list : ARRAY[1 : product_selling] OF STRING
16 DECLARE reserve_price_list : ARRAY[1 : product_selling] OF FLOAT
17 DECLARE item_number_list : ARRAY[1, product_selling] OF INTEGER
18
19 FOR counter <-- 1 TO product_selling
20     OUTPUT "Please enter your item name, item description and item price"
21     INPUT _name, _description, _price
22     name_list[counter] <-- _name
23     description_list[counter] <-- _description
24     reserve_price_list[counter] <-- _price
25 NEXT
```

Task 2

Allow buyers to purchase

```
1  FOR i 1 TO product_selling
2      highest_bid_list[i] <-- 0.0
3      item_highest_bid_holder_list[i] <-- ""
4  NEXT
5  FOR i 1 TO 6
6      buyer_number_list[i] <-- i
7  NEXT
8  buyer_number_check <-- "0"
9  cookie <-- False
10 WHILE purchase_status <-- "no" DO
11     OUTPUT "Welcome to auction! "
```

```

12     FOR counter_2 <-- 1 TO name_list
13         _item_num <-- item_number_list[counter_2]
14         current_item_name <-- name_list[counter_2]
15         OUTPUT _item_num, ": ", current_item_name
16     NEXT
17     CASE buyer_number_check OF
18         "0": OUTPUT "IF you want to bid, please enter your buyer number" INPUT
buyer_number_check
19         "exit": BREAK
20     ENDCASE
21     WHILE buyer_number_check IN buyer_number_list DO
22         IF NOT cookie
23             THEN OUTPUT "Identity verified."
24             INPUT item_to_buy
25         ENDIF
26         IF item_to_buy NOT IN name_list
27             THEN OUTPUT "Item number invalid, try again."
28             CONTINUE
29         ENDIF
30         search_index <-- name_list.index(item_to_buy)
31         current_description <-- description_list[search_index]
32         item_highest_bid <-- highest_bid_list[search_index]
33         OUTPUT "Details: ", current_description
34         OUTPUT "Current highest bit is ", item_highest_bid
35
36         REPEAT
37             INPUT buyer bid
38             IF buyer_bid > item_highest_bid
39                 THEN item_highest_bid <-- buyer_bid
40                 highest_bid_list[search_index] <-- item_highest_bid
41                 bid_number_list[search_index] <-- bid_number_list[search_index] + 1
42                 item_highest_bid_holder_list[search_index] <-- buyer_number_check
43                 OUTPUT "Congratulation! Your bid is the current highest bid."
44                 cookie <-- True
45                 BREAK
46             ELSE
47                 OUTPUT("Sorry, bid lower than current highest bid. Try again.")
48             ENDIF
49             INPUT purchase_status("Do you want to bid FOR another item? Y/N")
50         UNTIL purchase_status = "n"
51         IF buyer_number_check NOT IN buyer_number_list
52             THEN OUTPUT "Buyer number invalid, try again. "
53             buyer_number_check <-- "0"
54         ENDIF
55     ENDWHILE
56 ENDWHILE

```

Task 3

Calculate and show statistics

```

1 DECLARE highest_price_list : ARRAY[] OF INTEGER

```

```

2 DECLARE under_reserve_price_list : ARRAY[] OF INTEGER
3 DECLARE no_bid_list : ARRAY[] OF INTEGER
4 FOR i <-- 1 TO product_selling
5     sold_status_list[i] <-- ""
6 NEXT
7 total_price <-- 0
8 FOR counter_3 <-- 1 TO highest_bid_list
9     IF highest_bid_list[counter_3] = 0
10         THEN no_bid_list[counter_3] <-- counter_3
11         sold_status_list[counter_3] <-- "no"
12     ENDIF
13     IF highest_bid_list[counter_3] < reserve_price_list[counter_3] AND
highest_bid_list[counter_3] <> 0
14         THEN under_reserve_price_list[counter_3] <-- counter_3
15         sold_status_list[counter_3] <-- "no"
16     ENDIF
17     IF highest_bid_list[counter_3] > reserve_price_list[counter_3]
18         THEN highest_price_list[counter_3] <-- counter_3
19         sold_status_list[counter_3] <-- "yes"
20         total_price <-- total_price + highest_bid_list[counter_3] * 1.1
21     ENDIF
22 NEXT
23
24 OUTPUT "Total price is $", total_price
25 OUTPUT "following item has at least 1 bid, but the bid is lower than the reserve
price:", under_reserve_price_list
26 OUTPUT "following item has no bid at all: ", no_bid_list
27
28 sold_item_quantity <-- length(highest_price_list)
29 under_reserve_price_item_quantity <-- length(under_reserve_price_list)
30 no_bid_quantity <-- length(no_bid_list)
31 OUTPUT sold_item_quantity, " is/are sold."
32 OUTPUT under_reserve_price_item_quantity, " is/are lower than reserve price."
33 OUTPUT no_bid_quantity, " has/have no bids."

```