

## Programming Practice: Vending Machine

1. Suppose a vending machine sells the following eight products:

- A. Coca Cola, unit price US \$1.65
- B. Minute Maid Orange Juice, unit price US \$3.50
- C. Evian Mineral Water, unit price US \$2.80
- D. M&M's Chocolate, unit price US \$1.50
- E. Hershey's Chocolate Bar, unit price US \$1.85
- F. Oreo Cookies, unit price US \$1.00
- G. Doritos Tortilla Chips, unit price US \$3.25
- H. Pringles Potato Chips, unit price US \$3.40

Write a C program to simulate this vending machine with the following rules. The vending machine starts with the five operational mode as below:

- a. Deposit bill(s) or coin(s),
  - b. Select product(s),
  - c. Cancel a product,
  - d. Purchase product(s),
  - e. Abort transaction.
1. When button **a** is selected, the vending machine enters the money deposit state and the customer can deposit either bills or coins. Before each money deposit, enter **B** for bill deposit and **C** for coin deposit.
    - i. When the money deposit is **B**, the customer can deposit a 1-dollar, 5-dollar, 10-dollar, or 20-dollar bill by entering integer 1, 5, 10, or 20 to represent the value of a bill, respectively. The money deposit state ends when the value is 0. If the integer value is a number other than 0, 1, 5, 10, and 20, the money deposit is ignored, the vending machine stays in the money deposit state.
    - ii. When the money deposit is **C**, the customer can deposit a nickel (5 cents), a dime (10 cents), or a quarter (25 cents) using integer 5, 10, or 25 to represent the value of a coin, respectively. The money deposit state ends when the value is 0. If the integer value is a number other than 0, 5, 10, and 25, the money deposit is ignored and the vending machine stays in the money deposit state.
    - iii. If the money deposit is neither **B** nor **C**, the input is ignored and the money deposit state ends. The vending machine is back to wait for the next operational mode.
  2. When button **b** is selected, the vending machine enters the product selection state and the customer can select a product by entering one of characters **A**, **B**, **C**, **D**, **E**, **F**, **G**, and **H** as described above. In state **b**, a customer may select multiple items until he/she enters **Q** to quit; if any of the other characters is entered, it is ignored.
  3. When button **c** is selected, the customer can cancel a selected product item by entering one of **A**, **B**, **C**, **D**, **E**, **F**, **G**, and **H**. Each time a button **c** is selected, only one item can be canceled. If any of the other characters is entered, it is ignored. When a product is canceled, its quantity in the current transaction must be greater than 0 and the quantity is decreased by 1; otherwise, the cancellation has no effect.
  4. When button **d** is selected, the total price of the selected products is calculated.
    - i. If the deposited money is greater than or equal to the total price, print the selected products and quantities to simulate the vending machine dropping the selected products and return the changes to the customer. The returned changes contain coins only and must be the least number of coins. You may assume the vending

machine has sufficient product inventory, bills, and coins. A transaction is completed when the changes are returned. The vending machine may begin a new transaction.

- ii. If the deposited money is less than the total price, show a message "Insufficient Deposit" and enter state **a** to deposit additional money.
5. When button **e** is selected, the current transaction is canceled, return all the deposit money to customer. Assume the vending machine can return both bills and coins. Show the value and the number for each bill or coin deposited by the customer and the total amount returned. When the transaction is canceled, the deposited money is returned. The vending machine may begin a new transaction.

At the end of each of the above steps, show the *current transaction status* including the total deposit amount, the name and quantity of each selected product with its unit price, and the total price of the selected products. Program solution: `vending_machine.c`.

2. Redo vending machine using data structure **struct**. Program solution: `vending_machine_structure.c`.