

06 Hr 00 Min
54 Sec

Guidelines

Coding Area

Public Testcase
Submissions

Private Testcase
Submissions

Unevaluated
Submissions

Feedback Form

Graphs

Coding Area

A**B****C****D****E****F****ONLINE EDITOR (A)**

Codekart

+ Problem Description

Codu wants to create a shopping application. The application would sell only SHIRT and SHOE and have a cost that can be modified based on market needs. This application should allow users in two roles, viz. store manager(SM) and shopper(S).

Codu wants to test the app. He wants the application to execute a few commands and print the output.

Following is the list of allowed **commands**:

CMD SM ADD [ITEM NAME] [ITEM QTY] - adds the given quantity of the item to the inventory and prints(returns) the quantity added, otherwise prints -1 when there is any error or invalid input. Item qty can only be whole numbers > 0.

CMD SM REMOVE [ITEM NAME] - removes the item from the inventory, returns and prints -1 when there is an error, otherwise prints(returns) 1.

CMD SM GET_QTY [ITEM NAME] - returns and prints the currently available quantity for the item in the inventory, otherwise prints(returns) 0 in case the item is not found.

CMD SM INCR [ITEM NAME] [ITEM QTY] - adds the given quantity of the item to the inventory and prints(returns) the quantity added, otherwise prints(returns) -1 when there is any error or invalid input. item qty can only be whole numbers > 0.

CMD SM DCR [ITEM NAME] [ITEM QTY] - removes the given quantity of the item from the inventory and prints(returns) the quantity added, otherwise prints(returns) -1 when there is any error or invalid input. item qty can only be whole numbers > 0.

CMD SM SET_COST [ITEM NAME] [COST] - sets the cost of the item, returns the value, otherwise prints -1 in case of any errors or invalid input. Cost must be decimal.

CMD S ADD [ITEM NAME] [ITEM QTY] - adds the given quantity of the item to the shopping cart and prints(returns) the quantity added, otherwise prints -1 when there is any error or invalid input. Item qty can only be whole numbers > 0.

CMD S REMOVE [ITEM NAME] - removes the item from the shopping cart, returns and prints -1 when there is an error, otherwise prints(returns) 1.

CMD S INCR [ITEM NAME] [ITEM QTY] - adds the given quantity of the item to the shopping cart and prints(returns) the quantity added, otherwise prints(returns) -1 when there is any error or invalid input. item qty can only be whole numbers > 0.

CMD S DCR [ITEM NAME] [ITEM QTY] - removes the given quantity of the item from the shopping cart and prints(returns) the quantity added, otherwise prints(returns) -1 when there is any error or invalid input. item qty can only be whole numbers > 0.

CMD S GET_ORDER_AMOUNT - gets the total price of the items in the cart, returns the value, otherwise prints -1 in case of any error or invalid input. The total amount should be rounded and printed up to two decimal places.

NOTE- Increment and Decrement operations are only possible when the item is already in the inventory or cart. If increment or decrement is attempted on items that do not exist in the cart, then the command should return and print -1.

If an attempt is made to add an item that is already in the inventory or cart, such operations should result in an error and must return and print -1.

If an item which is present in the cart of inventory is removed using *remove* command and an increment or decrement operation is performed on it, such operations should result in an error and must return and print -1.

If any item quantity after decrement becomes zero, the same is removed from the corresponding inventory or cart. Performing increment or decrement operation after such a previous decrement operation, should result in an error and return -1.

You need to think of other similar error conditions while implementing the solution.

Please note at the beginning of a test case or command set, both the inventory as well as the cart is empty.

Here,

SM= STORE MANAGER

S= SHOPPER

You are required to create the application for Codu to manage the shopping kiosk.

The first line of input **T**, gives the number of test cases.

Each test set is a set of commands, which ends with "END" string.

Each command in a test case is on a new line

+ Constraints

$1 \leq T \leq 10$

+ Input

First line contains an integer T, which denotes the number of test cases

Second line onwards, there will be commands until we receive END command. Any command after the END command belongs to next test case.

For command format refer to Example section.

+ Output

Print output of every command. (Print double value for command SET_COST(rounding to one decimal places) and GET_ORDER_AMOUNT(rounding to two decimal places))

+ Time Limit

1

+ Examples

Example 1

Input

1

CMD SM SET_COST SHOE 5

CMD SM SET_COST SHIRT 10

CMD SM ADD SHOE 5

CMD SM ADD SHIRT 10

CMD SM DCR SHIRT 5

CMD SM INCR SHOE 5

CMD SM GET_QTY SHIRT

CMD SM GET_QTY SHOE

CMD SM REMOVE SHIRT

CMD SM GET_QTY SHIRT

CMD S ADD SHOE 2

CMD S INCR SHOE 2

CMD S DCR SHOE 1

CMD S GET_ORDER_AMOUNT

END

Output

5.0

10.0

5

10

5

5

5

10

1

0

2

2

1

15.00

Explanation :

From commands "**CMD SM SET_COST SHOE 5**" and "**CMD SM SET_COST SHIRT 10**"

We are successfully setting the cost as 5.0 and 10.0 respectively.

From next commands "**CMD SM ADD SHOE 5**" and "**CMD SM ADD SHIRT 10**"

Quantity of 5 shoes and 10 shirts has been successfully added to the inventory.

From next commands "**CMD SM DCR SHIRT 5**" and "**CMD SM INCR SHOE 5**"

Shirt quantity is decremented by 5 and shoe quantity is incremented by 5. This leaves us with 5 shirts and 10 shoes in the inventory.

From next commands "**CMD SM GET_QTY SHIRT**" and "**CMD SM GET_QTY SHOE**"

We are getting the quantity of shirt and shoe, which is 5 and 10 respectively.

From next command "**CMD SM REMOVE SHIRT**"

Shirt is removed from the inventory and hence 1 is printed.

From next command "**CMD SM GET_QTY SHIRT**"

We are querying the quantity of shirt, which is 0 as it was removed in the previous command.

From next command "**CMD S ADD SHOE 2**"

Shopper adds two shoes to the cart hence 2 is printed.

From next commands "**CMD S INCR SHOE 2**" and "**CMD S DCR SHOE 1**"

The user increments these shoes by 2 and then decrements by 1 hence 2 and 1 are printed. SO current shoes in cart= $2+2-1=3$

From next commands "**CMD S GET_ORDER_AMOUNT**"

The next command asks to print order amount or cart value, which is the cost of shoes * the number of shoes= $5*3=15$ hence 15.00 is printed by rounding to two decimal places.

Upload Solution [Question : A]

☐ I, **prajwal pasupuleti** confirm that the ☐ Took help from online sources
answer submitted is my own. (attributions)

Choose a
File...

[CodeVita FAQs](#)
[About CodeVita](#)
[Privacy Policy](#)
[Careers](#)

© 2020 Tata Consultancy Services Limited. All Rights Reserved.