You are to write a command-line cash machine (ATM).

The cash machine is pre-stocked with the following denominations:
• $100 - 10 Bills
• $50 - 10 Bills
• $20 - 10 Bills
• $10 - 10 Bills
• $5 - 10 Bills
• $1 - 10 Bills

Your application should take input from a console application and support the following commands:

• R - Restocks the cash machine to the original pre-stock levels defined above
• W<dollar amount> - Withdraws that amount from the cash machine (e.g.W $145)
• I<denominations> - Displays the number of bills in that denomination present in the cash machine (e.g. I $20 $10 $1)
• Q - Quits the application

The withdrawals from the cash machine should dispense cash in the most efficient manner possible, with the least amount of bills. After a withdrawal, the program should display success or failure and the remaining balance in the cash machine (sample output below). For an inquiry, the program should display the number of bills in the denominations specified (sample output below). After a restock, the program should display the balance in the cash machine (same as after a withdrawal). If the input is not understood, “Invalid Command” should be displayed. No additional messages, prompts or errors should be displayed.

Sample input/ouput. Note that the > are only in the sample to denote input and would not actually be a part of the problem as no additional prompts should be displayed.

> W $208
Success: Dispensed $208
Machine balance:
$100 - 8
$50 - 10

$20 – 10
$10 - 10
$5 - 9
$1 - 7

> W $9
Success: Dispensed $9
Machine balance:
$100 - 8
$50 - 10
$20 - 10
$10 - 10
$5 - 8
$1 - 3

> W $9
Failure: insufficient funds

> I $20 $1 $100
$20 - 10
$1 - 3
$100 – 8

> R
Machine balance:
$100 - 10
$50 - 10
$20 - 10
$10 - 10
$5 - 10
$1 - 10

> K
Failure: Invalid Command

You may use the language of your choice to implement the solution. Providing unit tests is encouraged, but not required. No external files or databases are required (i.e. the initial cash machine amount and restock levels can be hard-coded).