

Using std::map and std::pair

Write a program that defines a map<string, int>

Program asks user to enter a name or "stop" or "print". If user enters "stop" the program stops. If user enters "print" then the program prints the content of the map as key-value pairs. If input is neither "stop" nor "print" then program asks user to enter an integer. Then program uses the entered name as the key (~index to map) to store the entered integer.

To get the keys from the map you need to use iterators or range-for to loop through the map. When looping through the map the elements of the map are std::pairs that contain key and the value. If using iterator, the iterator "points" to a pair where first is key and second is the value. For example, if your iterator is called `it` then key of the current element is `it->first` and value is `it->second`.

For example:

```
Enter name or "stop" or "print": Keijo
Enter integer: 77
Enter name or "stop" or "print": Erkki
Enter integer: 23
Enter name or "stop" or "print": What
Enter integer: 5
Enter name or "stop" or "print": print
Erkki : 23
Keijo : 77
What : 5
Enter name or "stop" or "print": Himmel
Enter integer: 99
Enter name or "stop" or "print": print
Erkki : 23
Himmel : 99
Keijo : 77
What : 5
Enter name or "stop" or "print": Himmel
Enter integer: 55
Enter name or "stop" or "print": print
Erkki : 23
Himmel : 55
Keijo : 77
What : 5
Enter name or "stop" or "print": stop
<program stops>
```

Composition

Implement a class for calculating running average of a number of values. The idea of running average that average is calculated of last N values. For example, when running average of last 10 values is calculated every time a new value is added if there are already 10 values the oldest value is removed.

Implement a class called RunningAverage:

- Constructor takes number of values as a parameter for example:
`RunningAverage avg(5);`
Creates an instance that remember five last values.
- A deque of integers is used to store the values
- Class has member function `void add_value(int value)` which is used to add a new value and to remove the oldest value if there are specified number of elements so that number of values in the never exceeds the size given in the constructor.
- Class has member function `double get_average()` that returns the average of the stored values.

Write a program that creates running average object to calculate running average of last five numbers. Program asks user to enter numbers and to enter zero to stop. If entered value is not zero it is added to the running average and program prints the current running average. If value is zero the program prints the current running average and then stops.