Module 34.5 practise  
1.  
for(auto it=freq.begin(); it!=freq.end(); it++) cout << it->first << ": " << it->second << endl; explain this . why here not use \*it

In the line for(auto it=freq.begin(); it!=freq.end(); it++), we are creating an iterator it that points to the beginning of the freq map using freq.begin(). The iterator is then incremented using it++ in the for-loop until it reaches the end of the map, which is pointed by freq.end().

Now, to access the key-value pairs of the map, we use it->first to access the key (element) and it->second to access the value (frequency). This is equivalent to using (\*it).first and (\*it).second respectively.

So, we can use either it->first and it->second or (\*it).first and (\*it).second to access the key-value pairs of the map through the iterator it.

Using \*it directly to access the key-value pairs would result in a compilation error because \*it would dereference the iterator and return a reference to the key-value pair, not the key and value themselves. Therefore, we have to use either it->first and it->second or (\*it).first and (\*it).second.

1.2.

how the function works map<int, int> freq; // map to store frequency of each element for(int i=0; i<n; i++) freq[arr[i]]++;

Here's how the code works:

1. We create a map<int, int> called freq to store the frequency of each element in the array. Here, the key of the map is the element itself and the value is its frequency.
2. We loop through each element of the array using a for loop and increment the frequency of each element in the map using the following line:

freq[arr[i]]++;

This line uses the element arr[i] as the key to access the corresponding value in the freq map, and then increments the value by 1 using the postfix increment operator ++.

If the key arr[i] is not present in the map, it will be added to the map with a default value of 0, and then incremented by 1.

This effectively counts the frequency of each element in the array by incrementing the value associated with each element key in the map.

For example, consider the array {1, 2, 2, 6, 5, 5, 5, 6}.

After the first iteration of the loop (i=0), the map freq will contain the key-value pair {1: 1}.

After the second iteration of the loop (i=1), the map freq will contain the key-value pairs {1: 1, 2: 1}.

After the third iteration of the loop (i=2), the map freq will contain the key-value pairs {1: 1, 2: 2}.

And so on, until all elements of the array have been processed and the freq map contains the frequency of each element.

Note that the order in which the elements are processed in the loop does not matter, since the map data structure automatically sorts the keys in increasing order.