mohamedharraz21@gmail.com

# Hash table assignment

## Mohamed Ali Mohamed Harraz

# ID:4608

# Hash Table based On Chaining functions:

Put:

* Calculate the hash function given from the given image.
* Index=hash function(image).
* Node =Table[index]
* If node not NULL or available iterate with node =node->next until a node=null or node is available found.
* If node =NULL found node =malloc and the put the value and key in it.
* If node available found put the key and value in it.

Get:

* If Table[hash function]==NULL return not found.
* Node=Table[hash function].
* Else loop till node==NULL

If(node->value !=image->value)

Node=node->next

Else return node.

* Return not found.

Remove:

* If Table[hash function]==NULL return not found.
* Node=Table[hash function].
* Else loop till node==NULL

If(node->value !=image->value)

Node=node->next

Else clear the value and mark the key with available

flag

return node.

* Return not found.

# Hash Table based on linear probing functions:

Put:

* Calculate the hash function given from the given image.
* Index=hash function(image).
* If Table[index] not empty or available iterate with index =index+1 until an index is empty or available found.
* If an index is empty put the value and key in it and return.
* If loop 1000 times return tanle is full.

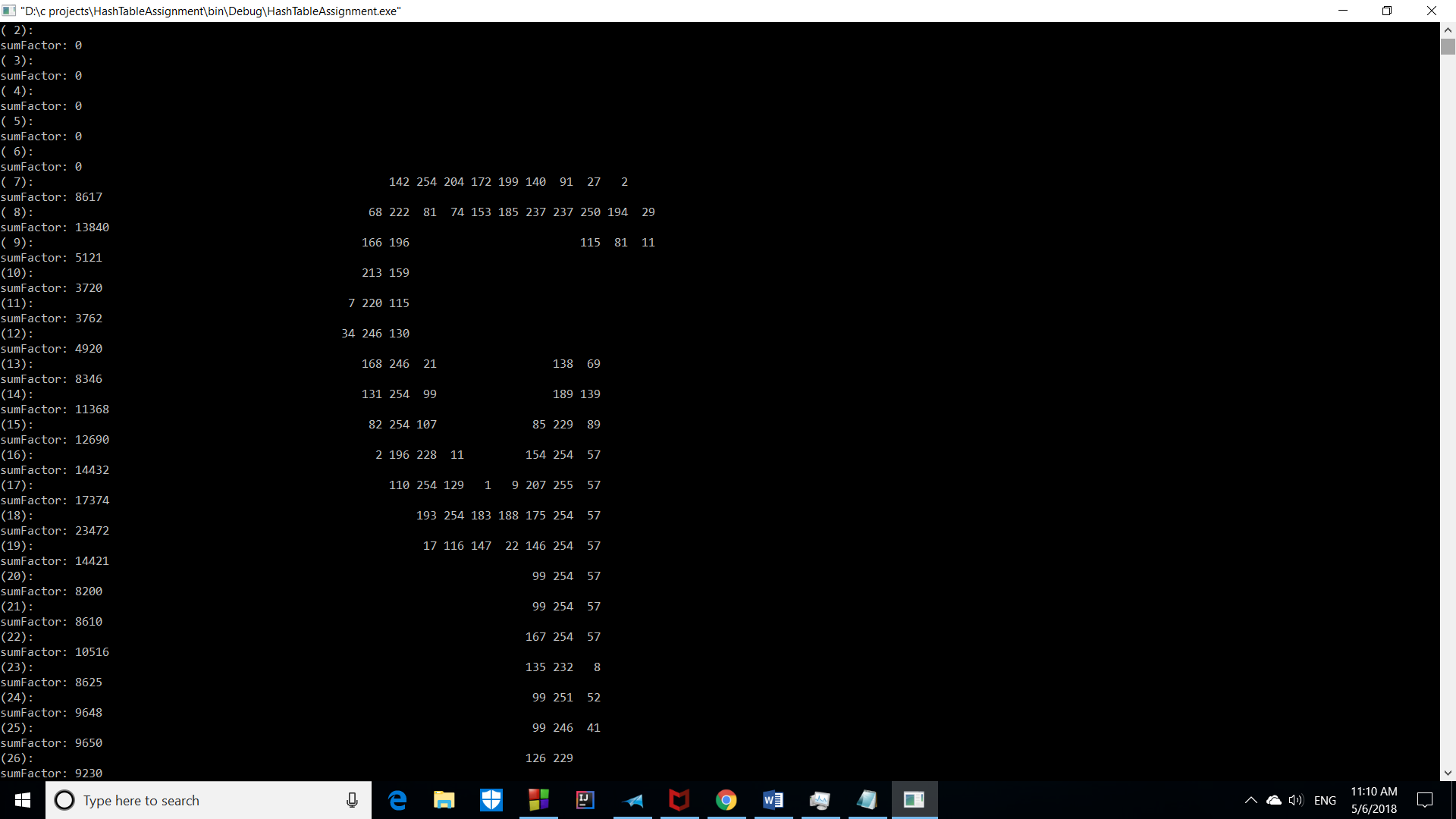
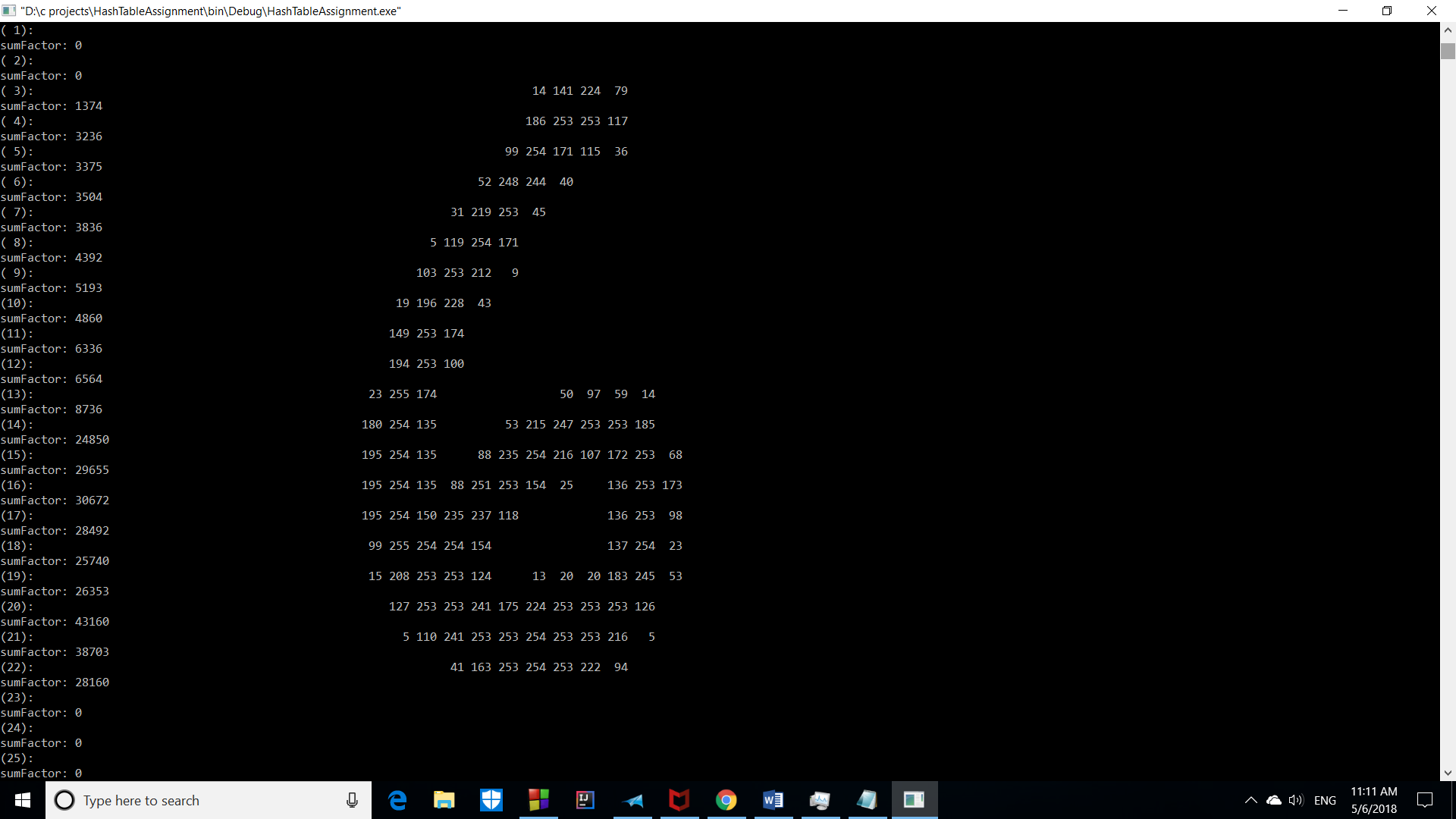
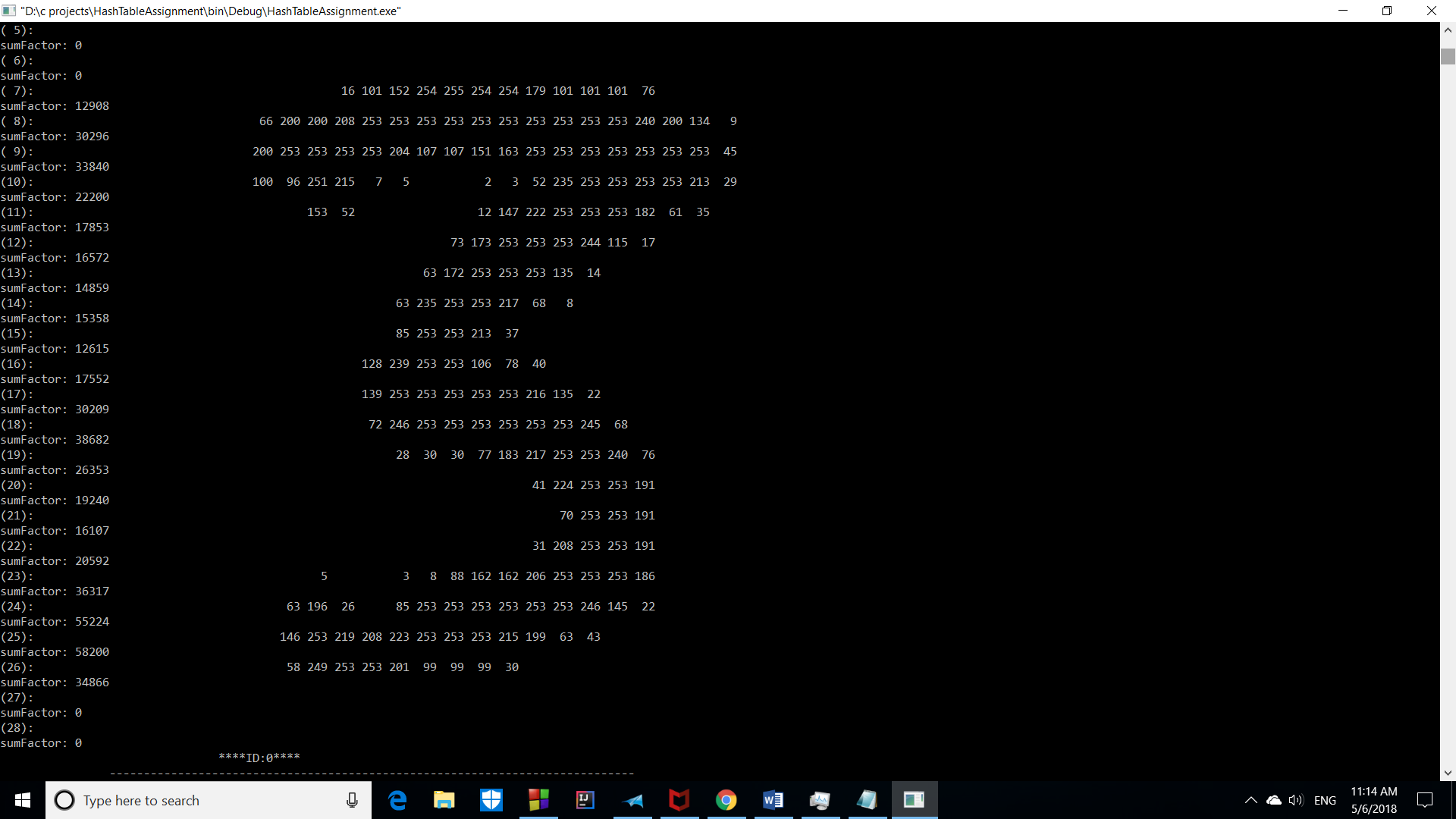
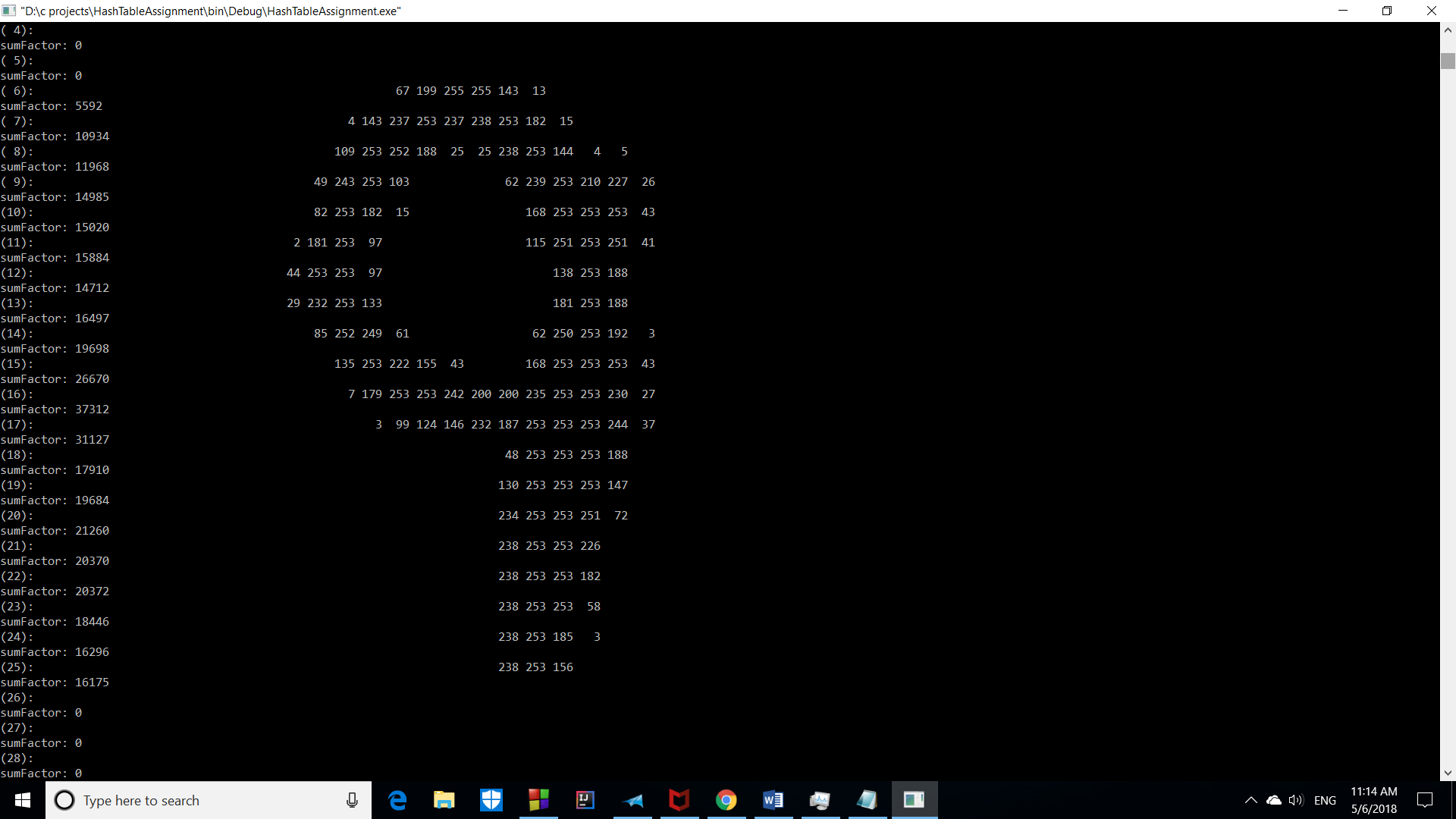
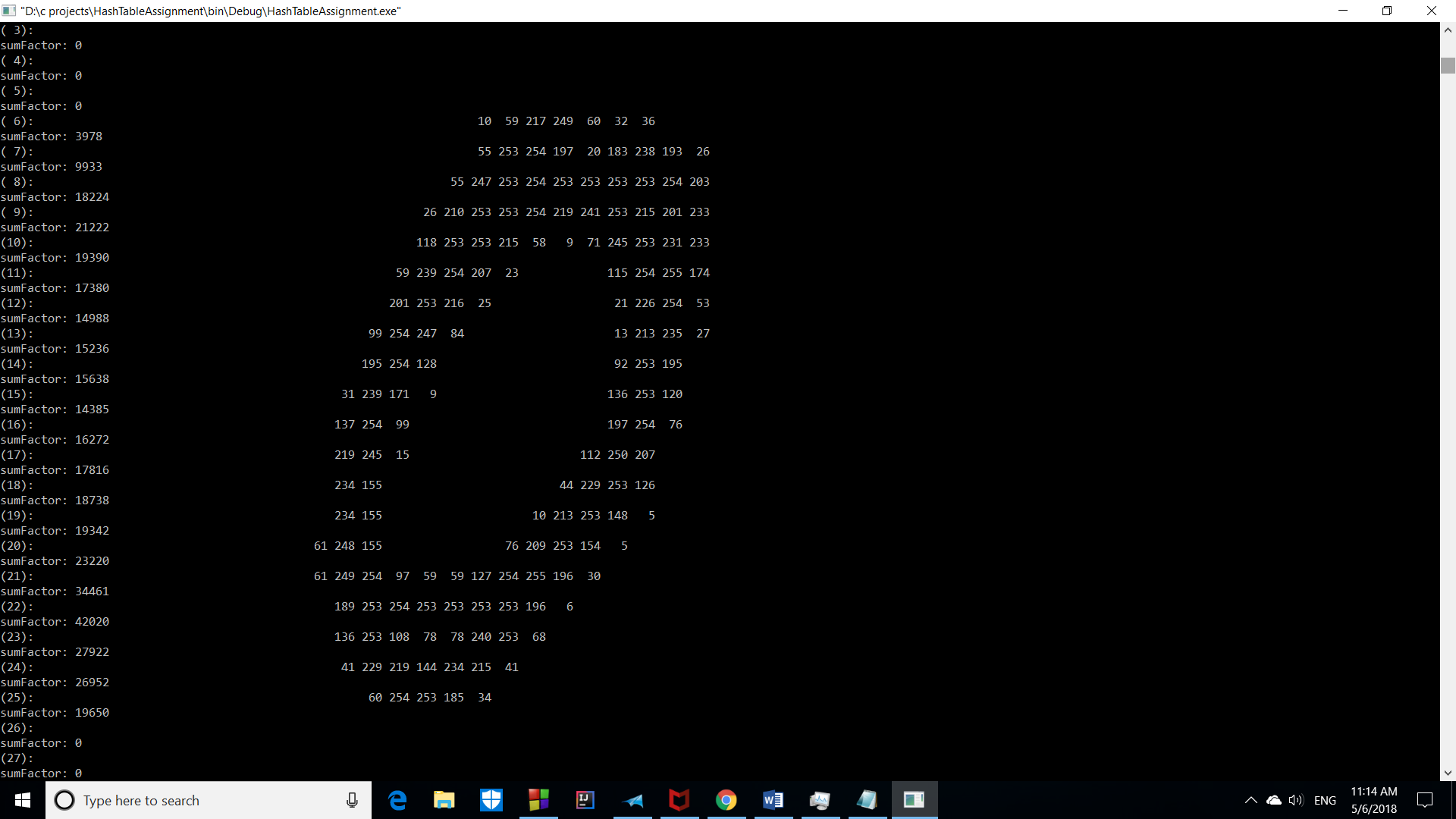
Get:

* Index =Hash function .
* If Table[index] empty return found.
* Else loop index=index+1 untill empty un available cell found or 1000 iterations reached compare values return key if found.
* Return not found.

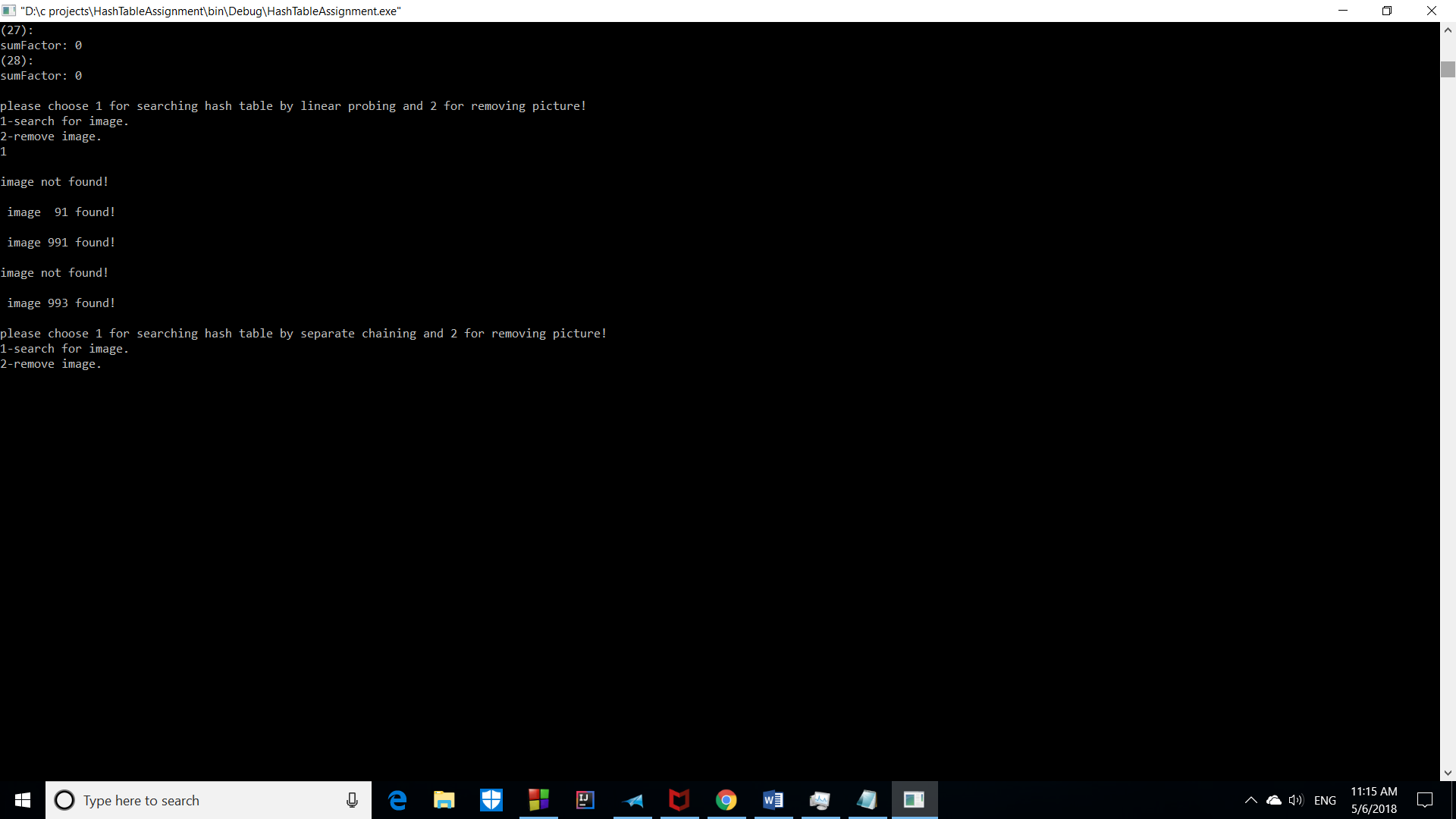
Remove:

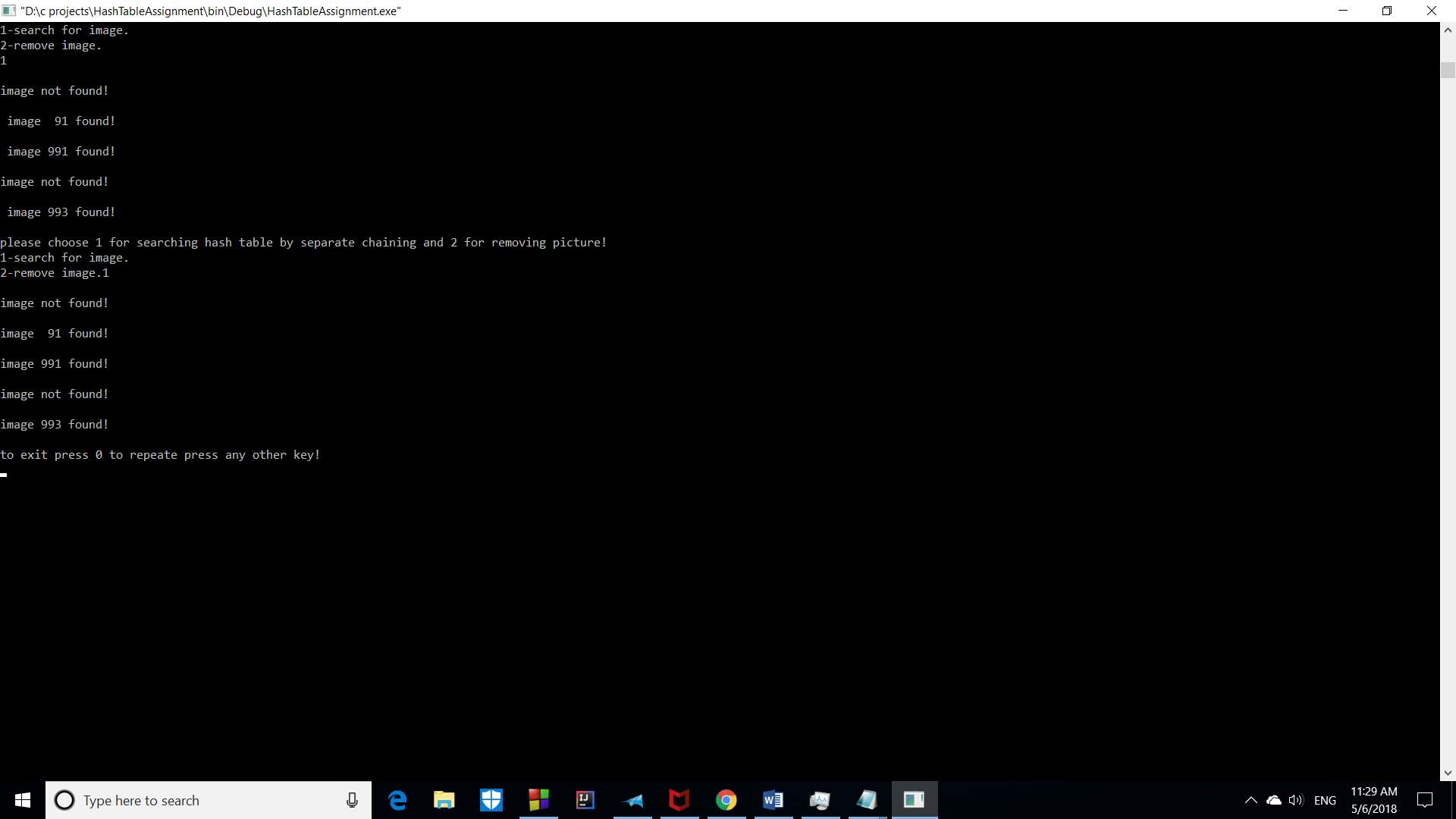
* Index =Hash function .
* If Table[index] empty return not found.
* If Table[index] .value ==image delete value mark key as available return key.
* Else loop index=index+1 untill empty un available cell found or 1000 iterations reached compare values delete value if found and mark available key return key if found.
* Return not found.

# Sample runs:

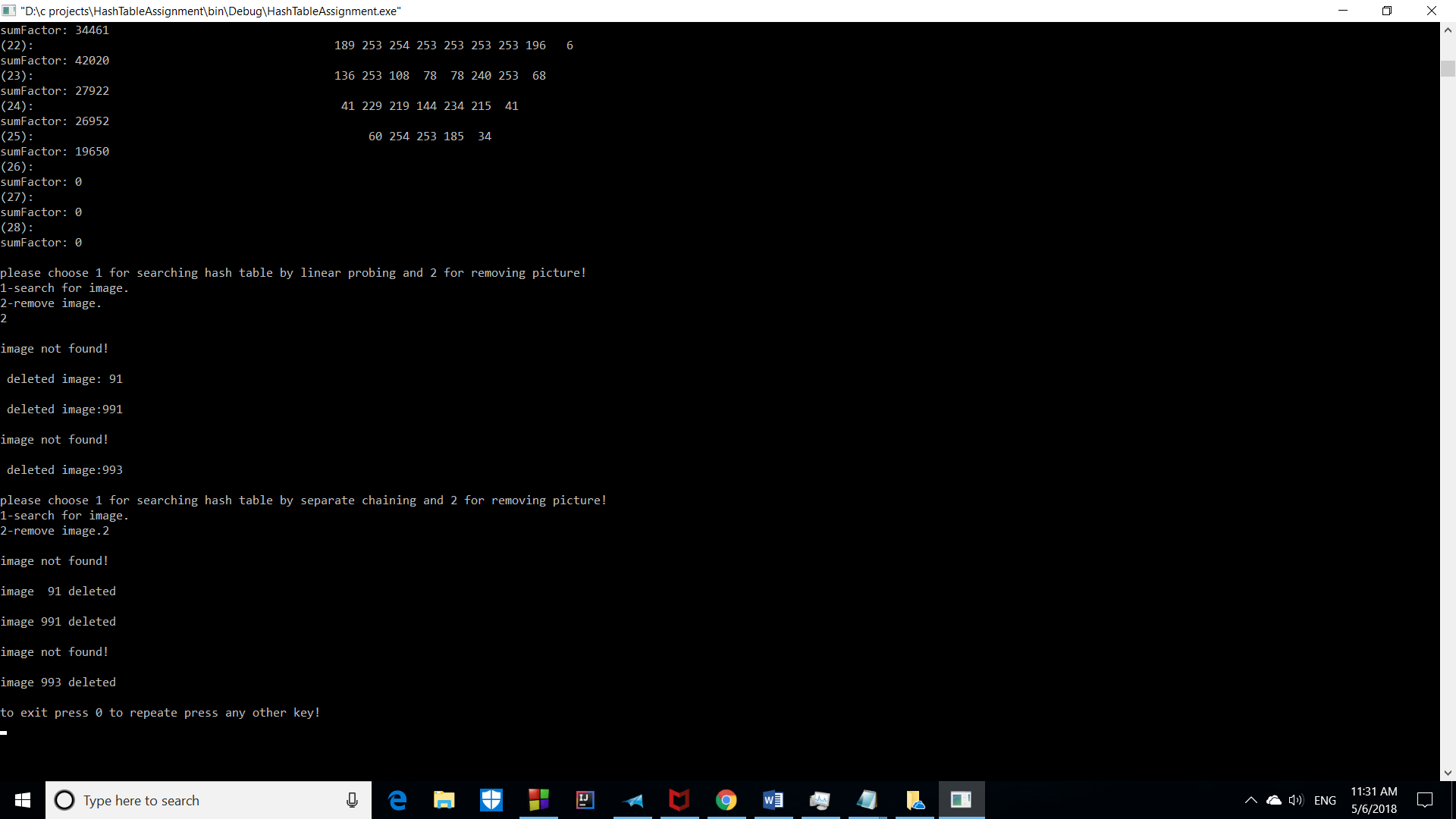
* Images loaded:
* 
* 
* 
* 
* 

# Searching:





Deleting:



# Collisions:

* chain collisions=378
* numbers of collisions in linear probing=499

# Time taken in inserting 1000 image by probing is 118507 milliseconds.

# Time taken in inserting 1000 image by separate chaining is 123839 milliseconds.