```cpp
#include <iostream>
#include <string>
using namespace std;

class HashTable
{
class Entry
{
public:
 int key;
 string name;
 Entry *next;

 Entry(int k,const string &s)
 {
  key = k;
  name = s;
  next = nullptr;
 }
};


public:
 Entry **table;
 int size;

public:
 HashTable(int s=10)
 {
  size =s;
  table = new Entry*[size];

  for (int i=0; i<size;i++)
   table[i] = nullptr;
 }

 int hash (int key)
 {
  int offset = key % size;
  return offset;
 }

 int put(int key,const string &name)
 {

  int offset = hash(key);

  if(table[offset] == nullptr)
  {
   Entry *newEntry = new Entry(key,name);
   table[offset]=newEntry;
   return 0;
```

```cpp
      }

      for (Entry *curr =table[offset];curr;curr= curr->next)
      {
       if(curr->key == key)
        {
         curr->name = name;
        }
        return 0;
      }



      Entry *newEntry = new Entry(key,name);
      newEntry->next=table[offset];
      table[offset]=newEntry;
      return 0;
    }


    bool get (int key,string &result)
    {
     int offset = hash(key);

     if (nullptr ==table[offset])
       return false;

     for (Entry *Curr= table[offset]; Curr;Curr = Curr->next)
     {
      if(Curr->key == key)
       {
        result =Curr->name;
        return true;
       }

     }
     return false;
    }

    void print()
    {
     for (int offset=0;offset<size;offset++)
     {
      cout<< "  " <<offset <<": ";
      for (Entry *curr=table[offset];curr;curr=curr->next)
       cout << "["<<curr->key << "]" << curr->name;

      cout <<endl;
     }
    }

};
```

```cpp
int main()
{
HashTable h1;
int key;
string result;

string name;
while (cout<<"Enter key(0 to stop) ", cin>>key,key)
{
cout<<" enter value ";
cin >>name;
h1.put(key,name);
}


h1.print ();

while (cout<<"Enter key to search(0 to stop)", cin>>key,key)
{
bool done = h1.get(key,result);
if(done)
cout << result<<endl;
else
cout<< "no key match" <<endl;

}

}
```