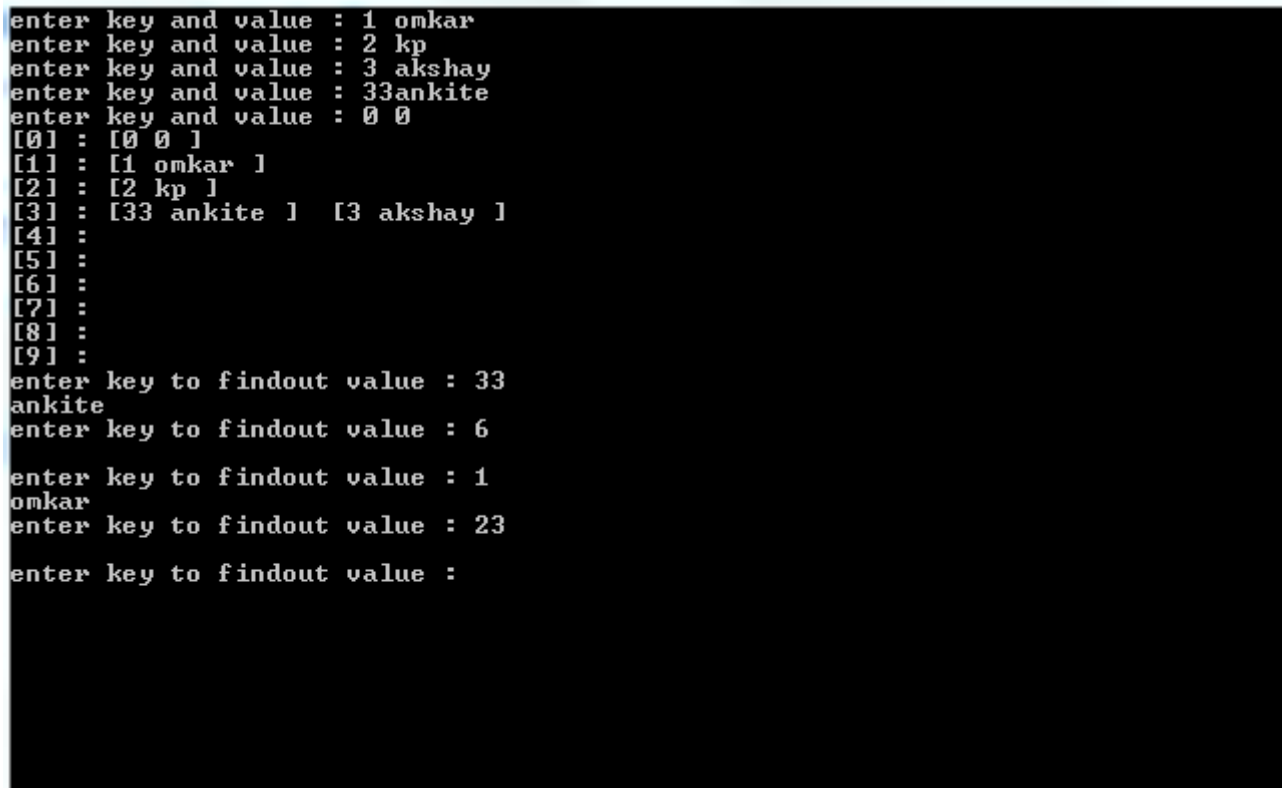
```cpp
#include <iostream>
#include <string>
using namespace std;

class Hashtable
{
	struct Entry
	{
		int key;
		string value;
		Entry* next;
		Entry(int id,string value):key(id),value(value),next(nullptr)
		{}
	};
	Entry **table;
	int size;
public:
	Hashtable(int size=10):size(size)
	{
		table=new Entry*[size];
		for (int i = 0; i < size; i++)
		{
			table[i]=nullptr;
		}
	}

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

	void put (int key, const string &value)
	{
		int offset = hash(key);

		for (Entry *current=table[offset]; current; current=current->next)
		{
			if (current->key == key)
			{

				current->value = value;
				return;
			}
		}

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

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

		for (Entry *current = table[offset]; current; current=current->next)
		{
			if(current->key==key)
			{
				result = current->value;
				return true;
			}
		}
		return false;
	}
```

```
        void printDebug ()
        {
                Entry *current=table[0];

                for(int i=0;i<size; i++)
                {
                        cout<<"["<<i<<"] : ";
                        for(Entry *current=table[i];current;current=current->next)
                        {
                                cout<<"["<<current->key<<" "<<current->value<<" "<<"]  ";
                        }

                        cout<<endl;
                }
        }

};

int main()
{
        Hashtable h;
        int key=1;
        string value="om";
        for(int i=0; key!=0 ;i++ )
        {
                cout<<"enter key and value : ";
                cin>>key;
                cin>>value;
                h.put(key,value);
        }


        h.printDebug();
        key=1;
        for(int i=0; key ;i++ )
        {
                cout<<"enter key to findout value : ";
                cin>>key;
                h.get(key,value);
                cout<<value<<endl;
                value=" ";
        }

        cin>>key;
}
```


enter key and value : 1 omkar
enter key and value : 2 kp
enter key and value : 3 akshay
enter key and value : 33ankite
enter key and value : 0 0
[0] : [0 0]
[1] : [1 omkar]
[2] : [2 kp]
[3] : [33 ankite] [3 akshay]
[4] :
[5] :
[6] :
[7] :
[8] :
[9] :
enter key to findout value : 33
ankite
enter key to findout value : 6
enter key to findout value : 1
omkar
enter key to findout value : 23
enter key to findout value :