25-11-2020

LAB-9

Program-8

NIHA
1BM18CS060

→ WAP to implement functions of Dictionary using Hashing :-

```
const int T_S = 200;
class HashtableEntry {
        public :
         int k; int v;
        HashtableEntry (int k, int v )
          { this → k = k;
            this → v = v; }
    };
class HashMapTable {
        private :
         HashTableEntry **t;
       public :
         HashMapTable() {
                t = new HashtableEntry *[T_S];
              for ( i=0; i<T_S ; i++)
                        t[i] = NULL ;
            }
          int HashFunc (int k)
           { return k % T_S; }
```

```
void Insert (int k, int v)
{
    int h = HashFunc(k);
    while (t[h] != NULL && t[h]->k != k)
        h = HashFunc(h+1);
    if (t[h] != NULL)
        delete t[h];
    t[h] = new HashTableEntry(k, v);
}
int SearchKey(int k)
{
    int h = HashFunc(k);
    while (t[h] != NULL && t[h]->k != k)
        h = HashFunc(h+1);
    if (t[h] == NULL)
        return -1;
    else
        return t[h]->v;
}
void Remove(int k)
{
    int h = HashFunc(k);
    while (t[h] != NULL)
    {
        if (t[h]->k == k)
            break;
        h = HashFunc(h+1);
    }
```

```
if (t[h] == NULL)
{
    cout << "No element found at key" << k ;
    return ;
}
else
{
    delete t[h]; }
cout << " Element Deleted " << endl;
}
~ HashMapTable()
{
    for ( i=0 ; i< T_S ; i++)
    {
        if ( t[i] != NULL)
            delete t[i];
            delete [] t;
    }
}
};
```