Akshay S Bharadwaj
1BM18CS011

ADS Lab-8 Writeup (Dictionary hashing)

```
const int MAX = 200;
class Hash~~table~~node {

    public:
        int key;
        int value;
        Hash~~table~~node (int key, int value) {
            this->key = key;
            this->value = value;

        }
};
class Dictionary {
    private:
        Dictionary **t;
    public:
        Dictionary () {
            t= new Hash~~table~~node *[MAX];
            for (int i = 0; i<MAX; i++)
                t[i] = NULL;
            }
        }
        ~Dictionary () {
            for (int i=0; i<MAX ; i++)
                if ( t[i] != NULL)
                    delete t[i];
                    detete [] t;
            }
        } ~~fa~~ int Hashfunc (int key);
        void insertion(int key ~~&~~, int value);
        ~~void~~ int search (int key);
        void deletion (int key);
};
```

Akshay S Bharadwaj

1BM18CS011

```
int Dictionary :: hashfunc (int key){
    return key % MAX;
}
void Dictionary :: insertion (int key, int value){
    int h = hashfunc(key);
    while (t[h] != NULL && t[h]->key != key){
        h = hashfunc(h+1);
    }
    if (t[h] != NULL)
        delete t[h];
    t[h] = new Hashnode (key, value);
}
int Dictionary :: search (int key) {
    int h = Hashfunc(key);
    while (t[h] != NULL && t[h]->key!= key){
        h = hashfunc (h+1);
    }
    if (t[h] == NULL)
        return -1;
    else
        return t[h] -> value;
}
void Dictionary :: deletion (int key){
    int h = hashfunc (key);
    while (t[h] != NULL) {
        if (t[h] ->key == key)
            break;
        h = hashfunc (h+1);
    }
    if (t[h] == NULL) {
        cout<< "No element at key \n"<< key << endl;
        return }
    else { delete t[h]; }
    cout << "Element has been deleted \n";
}
```