30/9/20 Lab 2:- Skip List

Abhiram G
1BM18CS127

```
int randomLevel () {
 float r = (float) rand() / RAND_MAX;
 int lvl = 0;
 while (r < P && LVl < MaxLVL)
 { lvl++;
    r = (float) rand() / RAND_MAX;
  } return Lvl;
};

void insertElemen (int key) {
 Node * current = header;
 Node * update [MAX LVL+1];
 memset (updak, 0, sizeof(Node*) * (MAXLVL + 1));
 for (int i = level ; i >= 0; i--) {
  while (current -> forward [i] != NULL &&
        current -> forward [i] -> key < key)
      Current = current -> forward [i];
  update [i] = current;  }
  current = current -> forward [0];
  if (current == NULL || current -> key != key) {
  int rlevel = randomLevel_ ();
   if (rlevel > level) {
    for (int i = level+1; i < rlevel +1; i++)
      updak [i] = header;
    level = rlevel;  }
  Node * n = CreateNode (key, rlevel);
  for (int i = 0; i <= rlevel ; i++) {
     n -> forward [i] = update [i] -> forward [i];
     update [i] -> forward [i] = n ; }
  cout << "successfully Inserted key" << key << "\n";
  } };
```

```
void deleteElement (int key) {
  Node * current = header;
  Node * update [MAXLVL+1];
  memset (update, 0, sizeof(Node*) * (MAXLVL+1));
  for (int i = level ; i >= 0; i--) {
   while (current -> forward [i] != NULL &&
          current -> forwad [i] -> key < key)
     current = current -> forward [i];
   update [i] = current ; }
 current = current -> forward [0];
 if (current != NULL and current -> key == key)
 {
    for (int i=0; i<=level ; i++) {
      if (updak [i] -> forward [i] != current)
         break;
      updat [i] -> forward [i] = current -> forward [i];
 } }
 while (level >0 && header -> forward [level] == 0)
  level --;
 cout << " Deleted " << key << "\n" } };

void  search fle (int key)
{
   Node * current = header;
   for (int i = level ; i >= 0; i--) {
   while (current -> forward [i] &&
          current -> forward [i] -> key < key)
     current = current -> forward [i] ; }
   current = current -> forward [0];
    if (current and current -> key == key)
      cout << "found:" << key << "\n ";
};
```