```
S.R. PUNEETH
                     Skip List placed
                                            1BMIECS087
cent cardon beroil () ?
float on = ( float ) rand ( ) / RAND-MAX ; while I would not be a float on a ( float ) and ( ) / RAND-MAX ;
int del = 0;
                        investiga (Noole 4. K her
while (ac P & & holk max did)
                      year & new near work a new work
  Jol++;
 er= (float) aand() / RAND_MAX;
word unset Eliour (int key) ( show the short
 Node & t current = chandre;
 Node + colate (MAX LVL+ 11); elout lindressis los
 menset (update, o, size of (Node k) + (MAXLVL+1)
  for (int i = dead; i>= 0; i--)
  while (correct -> forward (i)! = NULL & b
         annet -> forward (i) -> key ( key)
    Current = current -> forward (i)
   Update(i) = werent;
  cuant = current -> forward [0];
  inf (arguent = = NULL // augusent -> key! = key
   int a level = wardon tesel () . we ) sox) die
   if (or lovel > level)
    for(int i = devel+1; i < a dwel+1; i++)
     update [i] = cheader;
```

```
Wode & n = Create Noole (key, a houl);
for (int t=0; i == under devel; i++)
 n -> forward [i] = update [t] -> forward [i] in
 wpdate [i] -> forward [i] = n; }
 Cout << "Successfully arrested by "<< they ( ? " In")
 would delete Element (into key) {
 Node & arount: cheader
 Node * update (MAXLVL+1) juses bre dream) for
 menset (update, 0, sigeof (Noole *) * (MAXLVL+1)
 for (int i = divel; i>=0; i--) {
 while (arrent -) forward [i]! NULL &&
        correct => ( forward[i] -> key < key)
     current - cueent -> forward (i);
      update (i) = avenut; }
     consent: arrent -> forward (o ];
  of if (current! = NULL and current -> key = = key)
   for (int i=0; i <= devel; i++) {
   if (update [i] -, forward [i]! = werent)
    derenk
  update [i] - downed [i] = werest - forward [i];
  while (dword > 0 & & changes -> dornand (hold) = = 0)
  level - - ;
   cout << "Deleted" << key << "In" } };
```

word search file (at key) de loon et seed = 1 3 et al Node + aureurt = duaden;

for (int i = dual); i>= 0; i+)

while (aureut -> forward [i] &l

aureut -> forward [i] -> dey & < dey) everent = crocent > forward (i) jetelel him aif (correct and correct -> key = = key) to book

cout << "found" << key << " In " its law to be while (crossent -) forward [[] : NULL && count => com forward[i] => key < key) consult: cusunt -> forward (i); agdate (i) = would ; g consist: current -> forward (0); of if (surrent! : NUCL and would -> key = = dey) for Cink (30) is = dured; i++) { if (update [i] - forward [i]! = consert) update [:] - demound [:] - warest - forward [:] while (dued > 0 k & duader - downard (least) = = 0)