

1)

```
Function find_hash1(w,n)
    len=strlen(w)
    if(len==1)
        index=(2*w[0])%n
    Else
        index=((w[0]+w[1]+w[n-2]+w[n-1])/len)%n
    Return index
```

```
Function find_hash2(w,n)
    len=strlen(w)
    index=0
    for(i=0 to n)
        index=index+w[i]
    index=(index/len)%n
    Return index
```

```
Function store_word(H,n,w)
    index1=find_hash1(w,n)
    index2=find_hash2(w,n)
    Struct s well=createnode(w)
    count1=0
    count2=0
    Struct s a=H[index1]
    Struct s sel=H[index2]
    while(a.nxt!=NULL)
        a=a.nxt
        Count1++
    while(sel.nxt!=NULL)
        sel=sel.nxt
        Count2++
    if(count1<=count2)
        Struct s selli=H[index1]
        while(selli.nxt!=NULL)
            selli=selli.nxt
        selli.nxt=well
    Else
        Struct s sellis=H[index2]
        while(sellis.nxt!=NULL)
            sellis=sellis.nxt
        sellis.nxt=well
```

```

Function print_table(H,n)
    for(i=0 to n)
        if(H[i]==NULL)
            Print NULL
        Else
            Struct s sel=H[i]
            while(sel!=NULL)
                Print sel.word
                sel=sel.nxt

```

2)

```

Function read_and_store(ch,j,n)
    If j>n
        Return NULL
    flag=0, rto=0
    strncpy(sr,ch+j,ch+j+4)
    j=j+5

    While ch[j]!=' '
        num=ch[j]-'0'
        rto=(rto*10)+num
        J++
    ar[0]=0
    ar[1]=0
    root=newnode(rto,sr)
    index=-1
    If j<=n and ch[j+1]=='('
        index=findind(ch,j+1,n,ar)

    Return root

```

```

Function findind(ch,j,n,ar)
    If n<j
        Return -1
    z=0,stage=1,tot=0,flag=0,supreme
    For i=0 to n
        if(ch[i]=='(')
            Z++
            i=i+5
        Else if ch[i]==')'
            Z--

```

```

        Else if(isdigit(ch[i])
            k=ch[i]-'0'
            tot=tot*10 + k
        Else if ch[i]==' '
            If stage==1
                Ar[0]++
            Else if stage==2
                Ar[1]++

        tot=0
        If z==0 and stage==1
            supreme=i+1
            stage=2

    If ar[0]==0 and ar[1]==0
        Return -1
    Else
        Return supreme

Return -1

Function print_employees(T)
    if(T==NULL)
        Return NULL
    print_employees(T.left)
    Print T.eid
    print_employees(T.right)

Function find_employees(T)
    if(T==NULL||(T.left==NULL and T.right==NULL))
        return
    sum=findsum(T,sum)
    if(T.r>sum)
        Print T.eid
    find_employees(T.left)
    find_employees(T.right)

Function findsum(T,sum)
    if(T==NULL)
        Return sum
    sum=sum+T.r
    findsum(T.left,sum)
    findsum(T.right,sum)

```

