**Họ tên: Phan Công Trung**

**Mã sinh viên: 19T1021292**

Câu 1:

chưa lmà hết

Giải thuật 1

Đăt n = j+i;

T(n)

= {(C\_1 nếu i>j@2T(n/2)+C\_2 n nếu(i<j) )┤

T(n ) = 2⌊2T(n/4)+C\_2 n/2⌋ + C\_2 n = 4T n/4+2C\_2 n

T(n)

= 4[2T(n/8)+c\_2 n/4]+2C\_2 n = 8T n/8+3C\_2 n

…………………………

T(n)

= 2^i T(n/2i)+ic\_2 n

Quá trình sẽ kết thúc khi (n)/2^i = 1 hay 2i = n và do đó i = logn. Khi đó ta có:

T(n)

= nT(1) + lognC\_2n = C\_1n + C\_2nlogn = O(nlogn).

Giải thuật 2

Đăt n = j+i;

T(n) = T(n/2) +C\_1

T(n ) = T(n/2) +C\_1 + C\_1= T(n/2) + 2C\_1

T(n)

= T(n/2) +C\_1 + C\_1+ C\_1= T(n/2) + 3C\_1

…………………………

T(n)

= T(n/2i)+ic\_1

Quá trình sẽ kết thúc khi (n)/ 2^i= 1 hay 2i = n và do đó i = logn. Khi đó ta có:

T(n)

= C\_1 + logn = O(logn)

Câu 2:

1. struct SinhVien{

string hoten;

string lop;

string sbd;

float dtb;

SinhVien \*next;

};

2) a)

void Nhap\_SV(SinhVien \*&f, int n){

string hoten;

string lop;

string sbd;

float dtb;

for(int i=0; i<n; i++){

cout<<"Nhap ho ten sinh vien thu "<<i<<": ";

cin.ignore();

getline(cin, hoten);

cout<<"Nhap lop: ";

cin.ignore();

getline(cin, lop);

cout<<"Nhap so bao danh: ";

cin.ignore();

getline(cin, sbd);

cout<<"Nhap diem trung binh: ";

cin>>dtb;

SinhVien \*p = new SinhVien;

p->hoten = hoten;

p->lop = lop;

p->sbd = sbd;

p->dtb = dtb;

p->next = NULL;

if(f==NULL){

f = p;

}

else{

SinhVien \*q = f;

while(q->next != NULL)

q = q->next;

q->next = p;

}

}

}

b)

void xoaSinhVien(SinhVien \*&f){

if (f!=NULL){

SinhVien \*p=f;

while (p->next!=NULL){

if (p->next->dtb<5){

SinhVien \*q=p->next;

p->next = q->next;

delete q;

}

else p=p->next;

}

if (f->dtb<5){

SinhVien \*q = f;

f = f->next;

delete q;

}

}

}

Câu 3:

dạng cây:

1)Viết cài đặt cây bằng con trỏ

Typedef int item:

Struct Node {

Item key;

Node \*left;

Node \*right;

};

Typedef Node \*Tree;

2)

Thứ tự trước:

Void NLR(Tree T){

If(T!=Null){

Printf(‘%d “,T->key)

NLR(T->left);

NLR(T->right);

}

}

Thứ tự sau:

Void LRN(Node\*T){

If(T==Null){

LRN(T->left);

LRN(T->right);

Printf(‘%d “,T->key)

}

}

3)

+ Biểu thức tiền tố: +\*34/+12b6 => Biểu thức trùng với kết quả duyệt cây theo thứ tự trước

+ Biểu thức hậu tố: 34\*+12b+6/ => Biểu thức trùng với kết quả duyệt cây theo thứ tự sau