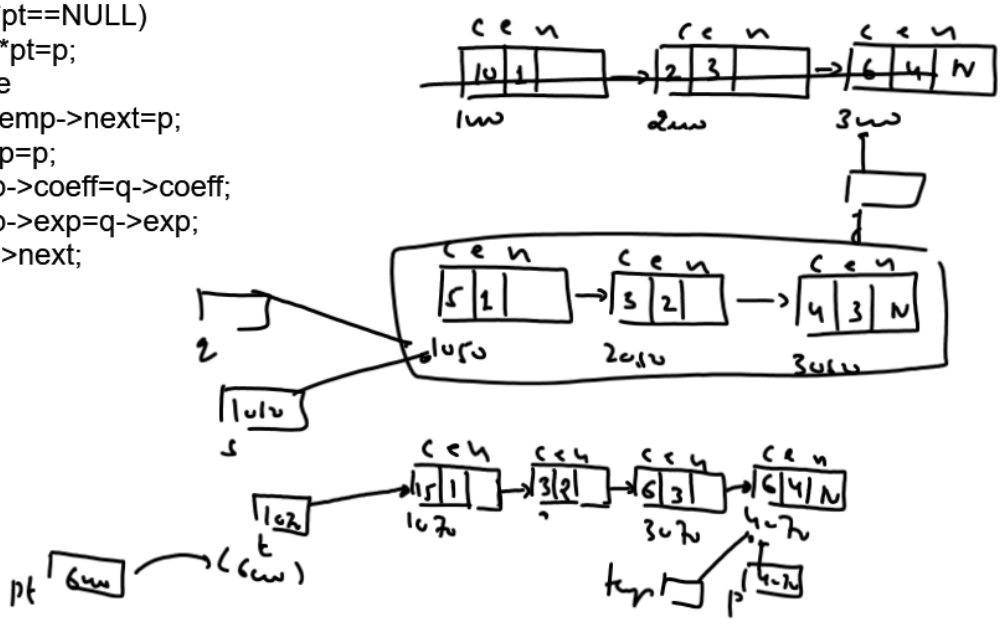
ADDITION OF POLYNOMIALS

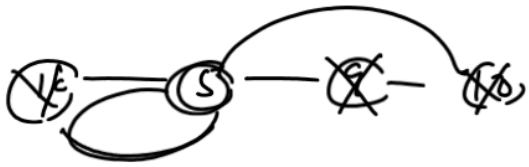
```
void add(struct poly *f,struct poly *s,struct poly **pt)
{
   struct poly *p,*temp,*q;
   if(f==NULL && s==NULL)
   {
      printf("Addition not possible");
      return;
   }
  while(f!=NULL && s!=NULL)
  {
     p=(struct poly *)malloc(sizeof(struct poly));
     p->next=NULL;
     if(*pt==NULL)
        *pt=p;
    else
       temp->next=p;
   temp=p;
   if(f->exp < s->exp)
    {
       temp->coeff=f->coeff;
       temp->exp=f->exp;
       f=f->next;
    }
   else if(s->exp < f->exp)
       {
         temp->coeff=s->coeff;
         temp->exp=s->exp;
         s=s->next;
       }
   else
     {
        temp->coeff=f->coeff+s->coeff;
        temp->exp=f->exp;
        s=s->next;
        f=f->next;
     }
}
if(f!=NULL)
   q=f;
else
   q=s;
```

```
   while(q!=NULL)
   {
        p=(struct poly *)malloc(sizeof(struct poly));
        p->next=NULL;
        if(*pt==NULL)
            *pt=p;
       else
           temp->next=p;
      temp=p;
     temp->coeff=q->coeff;
     temp->exp=q->exp;
     q=q->next;
   }
   }
```



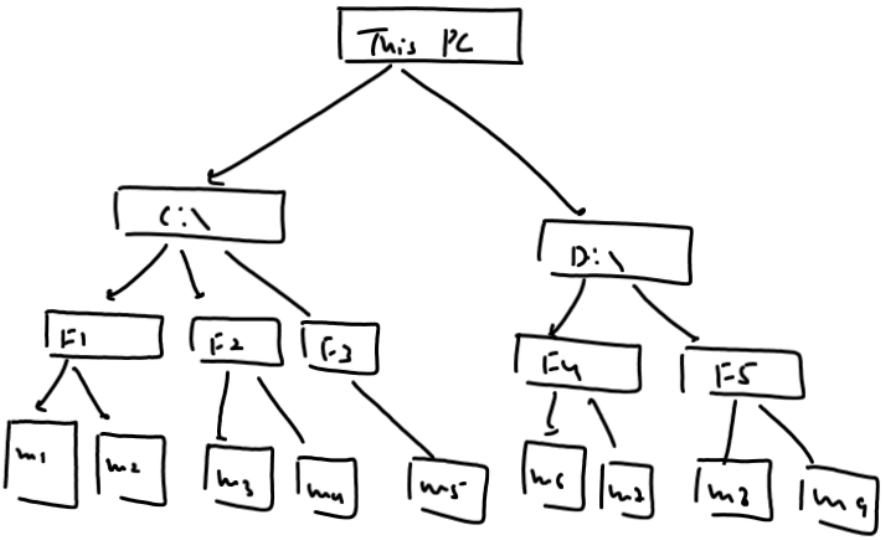
ASSIGNMENTS:
==========
Qn1. Modify the previous function add() of polynomial problem so that while adding new node add() calls append().

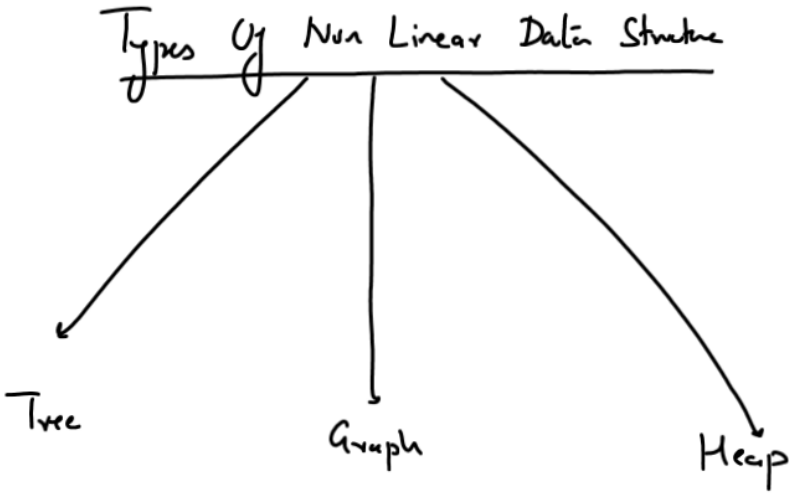
Qn2. Write down code to implement JOSEPHEUS PROBLEM



7

Non Linear Data Structure




Types Of Non Linear Data Structure
Tree
Graph
Heap