*Similarly the student Sn is registered for the classes C1, C2, C3

Polynomial differentiation:

Structure Declaration:

```
Void diff()
{
 Poly *ptr1, *newnode;
ptr1 = list1;
while (ptr1 != Null)
{
newnode = malloc(size of(struct poly));
newnode → coeff = ptr1 → coeff * ptr1 → exp;
newnode → exp = ptr1 → exp - 1;
newnode → next = NULL;
list3 = Create(list3, newnode);
ptr1 = ptr1 → next;
}
```

Routine:

```
Void Sub()
{
 Poly *ptr1, *ptr2, *newnode
 ptr1 = list1;
 Ptr2 = list2;
while (ptr1 != NULL || Ptr2 != NULL)
{
 newnode = Malloc (size of(struct poly));
 if (ptr1 → exp == ptr2 → exp)
{
newnode → Coeff = (ptr1 → coeff) - (ptr2 → coeff)); // Addition newnode → coeff = (ptr1 → coeff) + (ptr2 → coeff)
 newnode → exp = ptr1 → exp;
 newnode → next = NULL;
 list 3 = Create(list 3, newnode);
 ptr1 = ptr1 → next;
ptr2 = ptr2 → next;
{
```

```
else
{
 if (ptr1 → exp > ptr2 → exp)
 {
 newnode → coeff = ptr1 → coeff;
 newnode → exp = ptr1 → exp;
 newnode → next = NULL;
 list 3 = Create (list 3, newnode
 Ptr1 = ptr1 → next;
}
else
{
new → coeff = ptr2 → coeff;
new node → exp = ptr2 → exp;
newnode → next = NULL;
list 3 = Create (list 3, new node);
ptr2 = ptr2 → next;
}}}
```