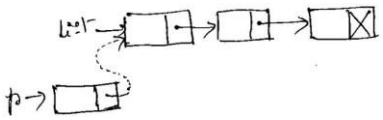
Realization of Singly Linked List:

(i) Insert an element into a list at the front.
(ii) Insert an element into a list at the last.
(iii) Insert an element into a list after a given node p
(iv) Delete an element from a list at the front.
(v) Delete an element from a list at the last.
(vi) Delete an element from a list after a given node p

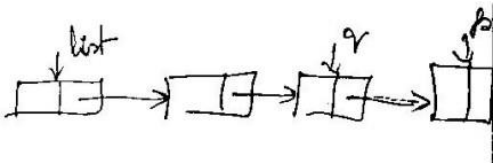
(i) Insert an element into a list at the front(left):

```
void insert_left(int x)
 {
    NODEPTR p;
    p = getnode();
    p→ info = x;
    p→ next = list;
    list = p;
 }
```



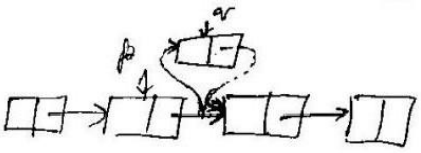
(ii) Insert an element into a list at the last(right):

```
void insert_end (int x)
{
NODEPTR p,q;
p=getnode();
p→info  = x;
p→ next = NULL;
if (list == NULL)
    list = p;
else
   {
    for(q=list; q→next!=NULL;q=q→next)
     {
        ;
     }
    q→next =p;
    }
}
```



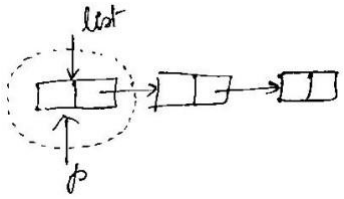
(iii) Insert an element into a list after a given node p:

```
void insert_after(NODEPTR p, int x)
{
    NODEPTR q;
    if(p==NULL)
      printf("Invalid insertion");
    else
      {
        q = getnode();
        q→info = x;
        q→next = p→next;
        p→next = q;
      }
}
```



(iv) Delete an element from a list at the front(left):

```
void delete_left()
{
NODEPTR p;
if(list == NULL)
printf("Invalid deletion\n");
else
{
p=list;
list = p→next;
p→next = NULL;
printf("Deleted element id %d", p→info);
free(p);
}
}
```



(v) Delete an element from a list at the last(right):

```
void delete_last()
{
   NODEPTR p,q;
    if(list == NULL)
    printf("Invalid deletion\n");
    p=list; q=NULL;
    while(p→next!=NULL)
    {
    q=p;
    p=p→next;
    }
    If(q!== NULL)
    q→next=NULL;
    else
    list == NULL;
    printf("Deleted element id %d", p→info);
    free(p);
}
```

(vi) Delete an element from a list after a given node p

```
void delete_after(NODEPTR p)
{
   NODEPTR q; int x;
   if (p==NULL||p→next == NULL)
       printf("Invalid deletion \n");
    else
        {
            q=p→next;
            x=q→info;
            p→next = q→next;
            q→next = NULL;
            free(q);
            printf(" Deleted element is %d", x);
         }
  }
```