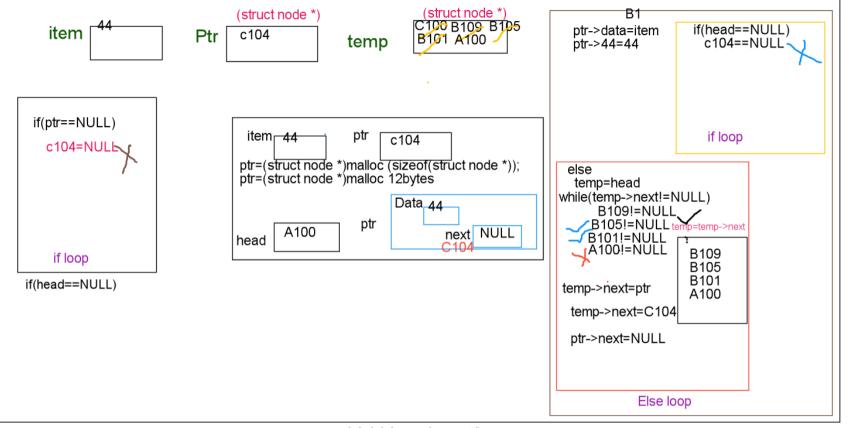
```
void beginsert()
    struct node *ptr;
                                                                item 99
                                                                                                                          item<sub> 88</sub>
                                                                                   ptr
                                                                                                                                            ptr
                                                                                         A100
                                                                                                                                                   B101
    int item:
    ptr = (struct node *) malloc(sizeof(struct node *));
                                                               ptr=(struct node *)malloc (sizeof(struct node *));
                                                                                                                        ptr=(struct node *)malloc (sizeof(struct node *));
    if(ptr == NULL)
                                                                                                                        ptr=(struct node *)malloc 12bytes
                                                               ptr=(struct node *)malloc 12bytes
                                                                                                                                                   Data 88
                                                                                         Data 99
       printf("\nOVERFLOW");
                                                                                                                                              ptr
                                                                                    ptr
    else
                                                                                                  next NULL
                                                                                                                                                            next A100
                                                               head
                                                                       A100
                                                                                                                          head B101
                                                                                                                                                            B101
       printf("\n Enter value\n");
                                                                                                         B10
       scanf("%d",&item);
       ptr->data = item;
       ptr->next = head:
                                                                                   ptr
                                                                                                                                             ptr
                                                                 item<del>. 77</del>
                                                                                                                          ∕item₁
                                                                                        B105
                                                                                                                                                   B109
       head = ptr;
                                                                                                                                 66
       printf("\nNode inserted");
                                                               ptr=(struct node *)malloc (sizeof(struct node *));
                                                                                                                          ptr=(struct node *)malloc (sizeof(struct node *));
                                                                                                                         ptr=(struct node *)malloc 12bytes
                                                               ptr=(struct node *)malloc 12bvtes
                                                                                                                                                    Data 66
                                                                                          Data_
                                                                                    ptr
                                                                                                                                               ptr
                                                                                                         B101
                                                                                                   next
                                                                                                                                                             next
                                                                        B105
                                                                head
                                                                                                                          head
                                                                                                                                  B109
                            88
                                    99
                                                                                                  B105
               66
                                                                  item<sub>□ 55</sub>
                                                                                          C100
                                                                 ptr=(struct node *)malloc (sizeof(struct node *));
                                                                ptr=(struct node *)malloc 12bytes
                                                                                          Data 77
                                                                                      ptr
                                                                                                    next B109
                                                                  head
                                                                           C100
```

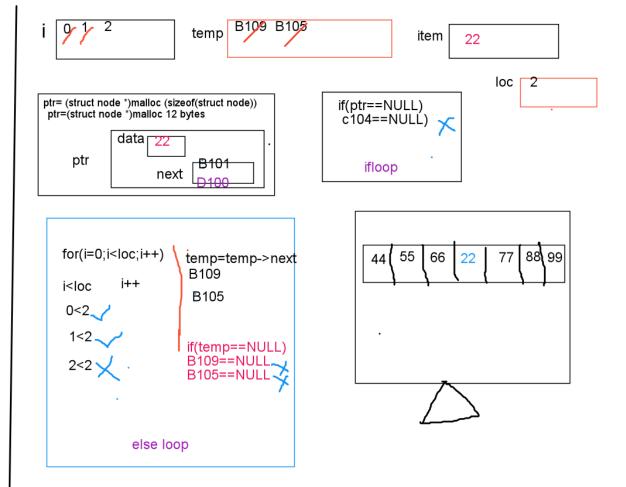
B105



66 77 88 99 last inserted element

Void Last insert()

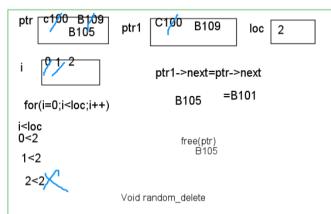
```
void randominsert()
     int i.loc.item:
     struct node *ptr, *temp;
     ptr = (struct node *) malloc (sizeof(struct node));
     if(ptr == NULL)
        printf("\nOVERFLOW");
     else
        printf("\nEnter element value");
        scanf("%d",&item);
        ptr->data = item:
        printf("\nEnter the location after which you want to insert ");
        scanf("\n%d",&loc);
        temp=head;
        for(i=0): i < loc: i++)
           temp = temp->next;
           if(temp == NULL)
             printf("\ncan't insert\n");
             return;
        ptr ->next = temp ->next;
        temp ->next = ptr;
        printf("\nNode inserted");
```

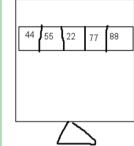


```
void begin delete()
     struct node *ptr
     if(head == NULL)
       printf("\nList is emptv\n"):
     else
        ptr = head
       head = ptr->next
       free(ptr)
       printf("\nNode deleted from the begining ...\n"):
                        else
                        ptr=Head
 if(head==NULL)
                          head=ptr->next
  A100==NULL) 🗸
                           head=NULL
                          free(ptr)
                           free(A100)
             voidbegin delete()
                                     88
               55 66
```

```
void last_delete()
     struct node *ptr *ptr1:
     if(head == NULL)
        printf("\nlist is empty"):
     else if(head -> next == NULL)
        head = NULL:
        free(head):
        printf("\nOnly node of the list deleted ...\n"):
      else
        ptr = head.
        while(ptr->next I= NULL)
          ptr1 = ptr
          ptr = ptr ->next
        ptr1->next = NULL:
        free(ptr)
        printf("\nDeleted Node from the last ...\n"):
       C180 B105 ptr1 C100 B109 B101
                                      if(head==NULL)
                        B185 B101
                                        c100==NULL
ifelse(head/>next==NULL)
    B109==NULL
                        else
                         while(ptr->next!=NULL)
                            B109!=NULL
                            B105I=NULL
                                           ptr1->next=Nul
                            B101!=NULL
                                             A100=NULL
                           A1001=NULL
                           NUII!=NULL
                                              free(A100)
```

```
void random_delete()
{
    struct node *ptr,*ptr1;
    int loc,i;
    printf("\n Enter the location of the node after which you want to perform deletion \n");
    scanf("%d",&loc);
    ptr=head;
    for(i=0,i<loc,i++)
    {
        ptr1 = ptr,
        ptr = ptr->next;
        if(ptr == NULL)
        {
              printf("\nCan't delete");
              return;
        }
        ptr1 ->next = ptr ->next;
        free(ptr),
        printf("\nDeleted node %d ",loc+1);
}
```





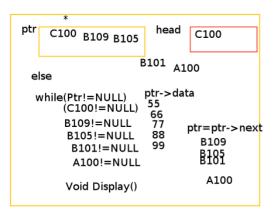
```
void search()
     struct node *ptr:
     int item,i=0,flag:
     ptr = head:
     if(ptr == NULL)
        printf("\nEmpty List\n");
     else
        printf("\nEnter item which you want to search?\n");
        scanf("%d",&item);
        while (ptr!=NULL)
          if(ptr->data == item)
             printf("item found at location %d ".i+1);
             flag=Ò:
           else
             flag=1;
           į++;
           ptr = ptr -> next:
        if (flag==1)
          printf("Item not found\n");
```

```
ptr
      6108104 A100 NULL i
                                0 1
        55
item
                         flag
                     else
                                          if(ptr->data==Item)
if(ptr==NULL)
                      while(Ptr!==NULL)
                                             55==55
(Č100==NUĹL\
                           C100!==NULL
                           B109!=NULL
                           B105!==NuLI
                                           ptr=ptr->next
                            B101!=NULL
                                           B109
                           A100!==NULL
                                            B105
                                            B101
                            NULL!=NULL
                                           A100
                                           NULL
```

Item found at 1 Location



```
void display()
    struct node *ptr:
    ptr = head;
     if(ptr == NULL)
       printf("Nothing to print");
     else
                                                             99
                                               77
                                                     88
                                       55 66
       printf("\n printing values\n ");
        printf("\n----\n"):
       while (ptr!=NULL)
          //printf("----");
          printf("| %d ",ptr->data);
          //printf("----"):
          ptr = ptr -> next:
        printf("|\n----\n"):
```



```
choice 0 1/1 1/1 1/2 3.4 5,6.7 8
   While(Choice!=9)
          (0!=9) case 5:
switch(choice)
                              last delete():
                              break:
           case 1:
           beginsert():
                              case 6:
           break:
                              random delete()
                              break;
           case 2:
           lastinsert():
                              case 7:
           break:
                               search():
                              break:
           case 3:
           randominsert():
                              case 8:
           break:
                              display();
                              break:
           case 4:
          begin delete();
                              case 9:-
           break:
                              exit(0):
```

```
Enter your choice:1
   Enter Value:99
   node inserted succesfully!
   Enter your choice:1
   Enter Value:88
  node inserted successfully!
  Enter your choice:1
  Enter Value:77
  node inserted successfully!
  Enter your choice:1
  Enter Value:66
  node inserted successfully!
  Enter your choice:1
  Enter Value:55
  node inserted successfully!
  Enter Your choice:2
  Enter Value: 44
 node inserted succesfully!
  Enter your choice:3
  Enter value:22
  Enter the loction after which you want inset
  Enter your choice:4
  node deleted beging...
  Enter your choice:5
  node deleted last...
Enter your choice:6
Enter the location of the node after which you want to perform deletion :2
deleted node is3
Enter your choice:7
enter the element you want serch:77
element found
  Enter your choice:8
                 77
                       88
                                99
          66
```