```c
//joseption problem//
#include<stdio.h>
#include<stdlib.h>
struct node
{
    char info;
    struct node* next;
};
struct node *getnode();
void insert(char ch);
 void joseption();
struct node *p,*r=NULL,*f,*q,*l;
int main()
{
    int i=0,c;
    char ch;
    while(i==0)
    {
        printf("\n1.insert\n2.joseption");
        printf("\nyour choie");
        scanf("%d",&c);
        switch(c)
        {
        case 1:
            printf("\nenter char");
            scanf("%c",&ch);
            insert(ch);
            break;
        case 2:
            joseption();
            break;
        default:
            printf("\nenter valid num");
        }
    }
}
struct node *getnode()
{
    p=(struct node*)malloc(sizeof(struct node));
    p->next=NULL;
    return p;
}
void insert(char ch)
{
    p=getnode();
    p->info=ch;
    if(r==NULL)
    {
        r=p;
        r->next=r;
        f=p;
    }
    else
    {
        r->next=p;
        r=p;
        r->next=f;
    }
    printf("%c",r->info);
}
 void joseption()
 {
    int z=1,n;
    char m,c,x;
    printf("\nchoose 0any element random::::");
    scanf("%c",&m);
```

```c
67        printf("\nenter any position random::::");
68        scanf("%d",&n);
69     while(r->info!=m)
70     {
71          r=r->next;
72     }
73     q=r;
74     p=r->next;
75     while(r->next!=r)
76     {
77          while(z!=n)
78          {
79               p=p->next;
80               q=q->next;
81               z++;
82          }
83          l=p;
84          x=l->info;
85          p=p->next;
86          q->next=p;
87          r=q;
88          free(l);
89          z=1;
90          printf("\n%c",x);
91     }
92 }
```