JOSEPHUS PROBLEM

Exp. No.:
AIM:

ALGORITHM:

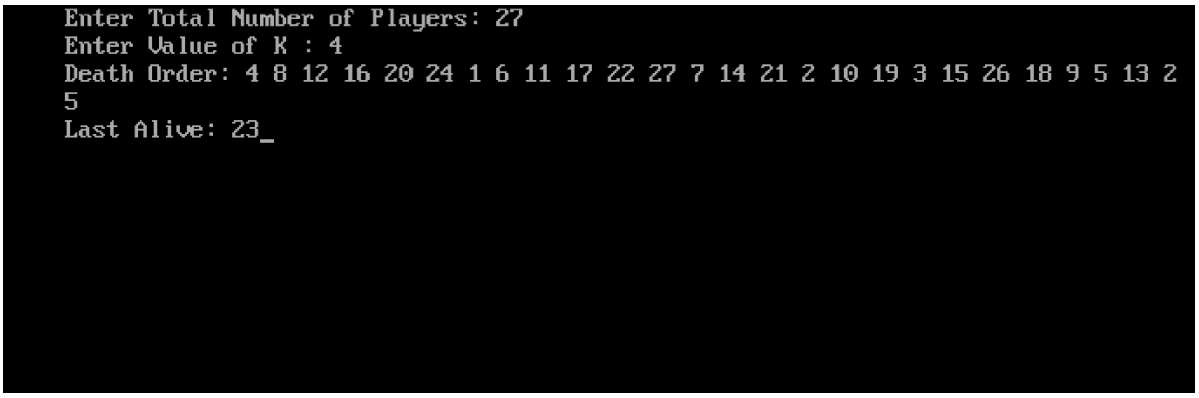
PROGRAM:

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

int i;

struct node
{
   int data;
   struct node *next;
};

void create(struct node **head, int n)
{
   struct node *temp, *prev;
   for (i = 1; i <= n; i++)
   {
      temp = (struct node *)malloc(sizeof(struct node));
      temp->data = i;
      if (i == 1)
      {
         *head = temp;
         prev = temp;
      }
      else if (i > 1)
      {
         prev->next = temp;
      }
      if (i == n)
      {
         temp->next = *head;
      }
      prev = temp;
   }
}
```

```
void traverse(struct node *head, int n)
{
   struct node *temp;
   temp = head;
   for (i = 0; i < n; i++)
   {
      printf("Data: %d \n", temp->data);
      temp = temp->next;
   }
}

void delete(struct node **data)
{
   struct node *temp;
   temp = *data;
   while (temp->next != *data)
   {
      temp = temp->next;
   }
   *data = (*data)->next;
   temp->next = *data;
}

void josephus(struct node *head, int k)
{
   struct node *temp;
   temp = head;
   printf("Death Order: ");
   while (temp->next != temp)
   {
      for (i = 1; i < k; i++)
      {
         temp = temp->next;
      }
      printf("%d ", temp->data);
      delete (&temp);
   }
   printf("\n");
   if (temp->next == temp)
```

```
    {
        printf("Last Alive: %d", temp->data);
    }
}

int main()
{
    struct node *head;
    int p, k;
    printf("Enter Total Number of Players: ");
    scanf("%d", &p);
    printf("Enter Value of K : ");
    scanf("%d", &k);
    create(&head, p);
    josephus(head, k);
    getch();
    clrscr();
    return 0;
}
```

OUTPUT:



RESULT: