Solution To Josephus Problem

Code Id	24
Date Updated	3/7/2010
Title	Solution to Josephus problem

Description

This is a program to solve Josephus problem. The Josephus problem consists of a g

Codes Snippet

```
#include
#include
struct node
{
        char name[20];
        struct node *next;
};
struct node *getnode(void)
{
        struct node *tp;
        tp=(struct node *)malloc(sizeof (struct node));
        printf(�n Input name n�);
        scanf(�%s�,tp->name);
        tp->next=NULL;
        return (tp);
}
void addnode(struct node **pplh, struct node *tp)
{
        struct node *travel;
        travel=NULL;
        if (*pplh=NULL)
        {
                *pplh = tp;
                (*pplh) -> next=*pplh;
        }


else
        {
                travel=*pplh;
                while(travel->next != (*pplh))
                        travel=travel->next;
                tp->next=*pplh;
                travel->next=tp;
        }
}
void create (struct node **pplh)
{
        struct node *getnode(void);
        void addnoce( struct node **pplh, struct node *tp);
        struct node *ltp;
        char ch1=�y�;
        ltp=NULL;
        while(ch1==�y�)
{
                ltp=getnode();
                addnode(pplh,ltp);
                printf(�nWant to add more names ?�);
                ch1=getchar();
        }
}
void countnode(struct node **pphl,int *lpn)
{
        struct node *travel, *tp;
        int ctr;
        tp=travel=NULL;
        travel=*pplh;
        if (*lpn==1)
        {
                while(travel->next !=*pplh)
```


Recent Posts

Types of Cloud Computing
What is Cloud Computing ?
How to pass a multi-dimensional array to a function?
Memory Layout of a C Program
PHP and Its Advantages

```
                        travel->next=travel->next->next;
                travel=travel->next;
                        free(tp);
                }
                printf(�nNAME LEFT IS %sn�,travel->name);
        }/*end else*/
}
main()
{
        struct node *getnode();
        void addnode (struct node **pphl,struct node *tp);
        void create (struct node *pphl);
        void countnode (struct node **pplh, int *lpn);
        struct node *phg;
        int n;
        phg=NULL;
        printf(�ninput the namesn�);
        create(&phg);
        printf(�ninput num to be moved�);
scanf(�%d�,&n);
        countnode(&phg, &n);
        getch();
}
```