**Computing Science 204 (B1)**

**The Josephus Problem**

Thursday, July 20

The following is an excerpt from the book *Concrete Mathematics, A Foundation for Computer Science*, by Graham, Knuth, and Patashnik.

> "Our final introductory example is a variant of an ancient problem named for Flavius Josephus, a famous historian of the first century. Legend has it that Josephus wouldn't have lived to become famous without his mathematical talents. During the Jewish–Roman war, he was among a band of 41 Jewish rebels trapped in a cave by the Romans. Preferring suicide to capture, the rebels decided to form a circle and, proceeding around it, to kill every third remaining person until no one was left. But Josephus, along with an unindicted co-conspirator, wanted none of this suicide nonsense; so he quickly calculated where he and his friend should stand in the vicious circle."

A working implementation of this algorithm is given below, in this version we start with $N$ people numbered 1 to $N$ arranged in a circle, and then we eliminate every $M^{\mathrm{th}}$ remaining person until only one survives.

```
#include <stdio.h>

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

main()
{
   int i, N, M;
   struct node *t,*x;
   printf("Enter N:\n");
   scanf("%d",&N);
   printf("Enter M:\n");
   scanf("%d",&M);
   t = (struct node *) malloc(sizeof *t);
   t->key = 1;
   x = t;
   for (i = 2; i <= N; i++)
   {
       t->next = (struct node *) malloc(sizeof *t);
       t = t->next;
       t->key = i;
   }
   t->next = x;

   while (t != t->next)
   {
       for (i = 1; i < M; i++)
           t = t->next;
       printf("%d ",t->next->key);
       x = t->next;
       t->next = t->next->next;
       free(x);
   }
   printf("%d\n",t->key);
}
```