

# 圆桌顺序

姓名：游霄童 学号：21009200158

created: 2023/10/19

## 问题描述

设有 $n$ (设为17)个人围坐在圆桌周围，按顺时针给他们编号(1, 2, ...,  $n$ )，从第1个人开始按顺时针方向加1报数，当报数到 $m$ (设为11)时，该人出列，余下的人继续进行，直到所有人出列。编写程序模拟这一过程，求出出列人的编号顺序。

## 一、基本原理及步骤（或方案设计及理论计算）

该问题通过使用汇编语言中的循环队列（Circular Queue）解决。循环队列是一种环形数据结构，适用于环形操作，例如人围坐在圆桌周围按照顺时针报数的问题。报数到 $m$ 时，出列一人，直到所有人出列。

### 步骤：

- 初始化：**设置初始指针、计数器，创建一个长度为 $n$ （17）的循环队列（QUEUE数组）。
- 循环报数和出列：**使用循环结构模拟人依次报数的过程，满足条件时出列，更新队列指针和计数器。
- 记录出列人的编号：**将出列人的编号记录在结果数组中。
- 结束程序：**当所有人出列后，结束程序。

这段汇编代码的目标是模拟一个问题：有17个人围坐在圆桌周围，按顺时针报数，当报数到11时，该人出列，直到所有人出列。以下是代码的简要步骤：**初始化：**数组VAR1存储1到17的整数，表示17个人的编号。创建一个长度为17的队列（QUEUE）用于存储出列人的编号。**报数和出列：**使用寄存器BX来遍历VAR1数组，模拟每个人的编号。使用寄存器AL来表示当前的报号，初始为1。当报号到达11时，将该人的编号存入QUEUE，然后重新报号（AL重新置为1）。遍历完所有人后，程序结束。

## 二、代码

```
STACK SEGMENT STACK
```

```
STACK ENDS
```

```
DATA SEGMENT
```

```
VAR1 DB 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 ;VAR1存储n
```

```
QUEUE DB 17 DUP(0)
```

```
DATA ENDS
```

CODE SEGMENT

ASSUME CS:CODE,DS:DATA,ES:DATA,SS:STACK

START:

MOV AX,DATA

MOV DS,AX

;BX为队列中的人的编号，AX为当前的报号

XOR AX,AX

XOR BX,BX

XOR CX,CX

XOR DX,DX

MOV CL,VAR1

MOV DX,DI

MOV AX,1

MOV SI,0

LOOP1:

CMP BL,11H

JNZ GO

MOV BX,0

GO:

MOV CL,[BX]

CMP CL,0

JNZ LOOP2

INC BX

JMP LOOP1

LOOP2:

CMP AL,0BH ;出循环的条件，AL为当前报数

JZ LOOP3

INC AL

INC BX

JMP LOOP1

LOOP3:

MOV [SI+17],CL

MOV [BX],0

MOV AX,1

CMP SI,10H

JZ OVER

INC SI

INC BX

JMP LOOP1

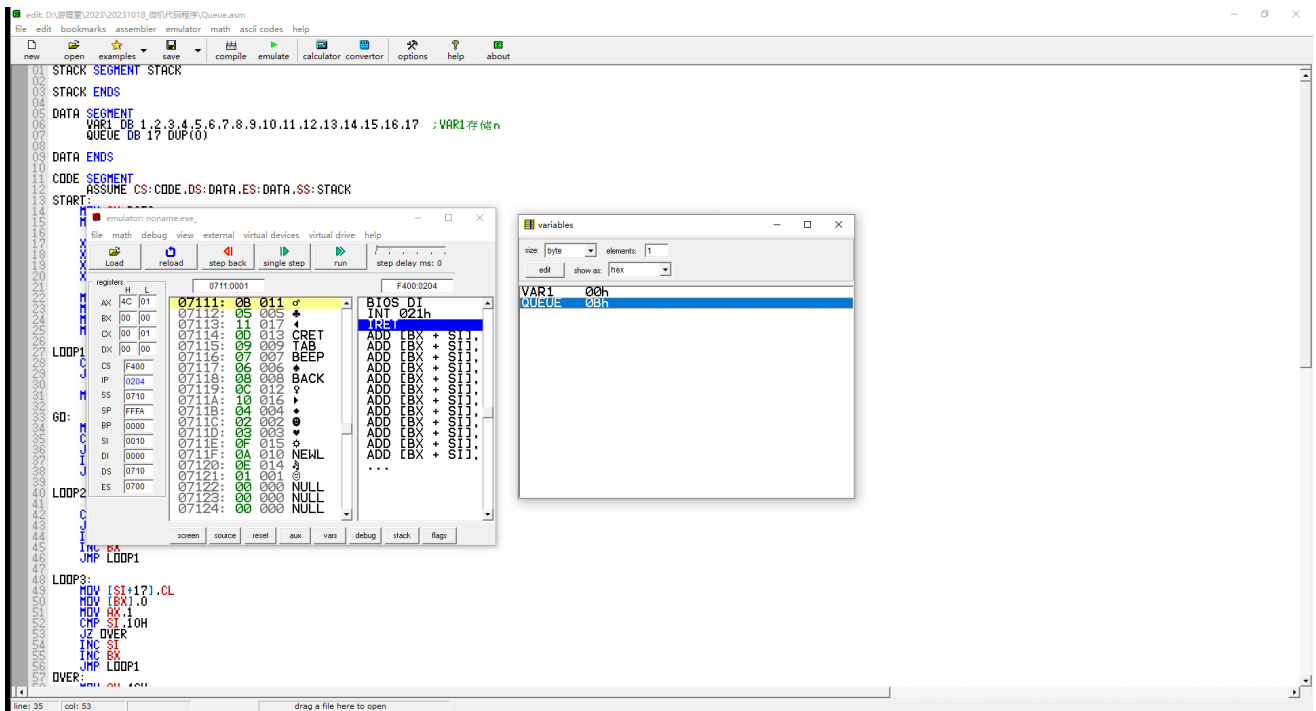
OVER:

MOV AH,4CH

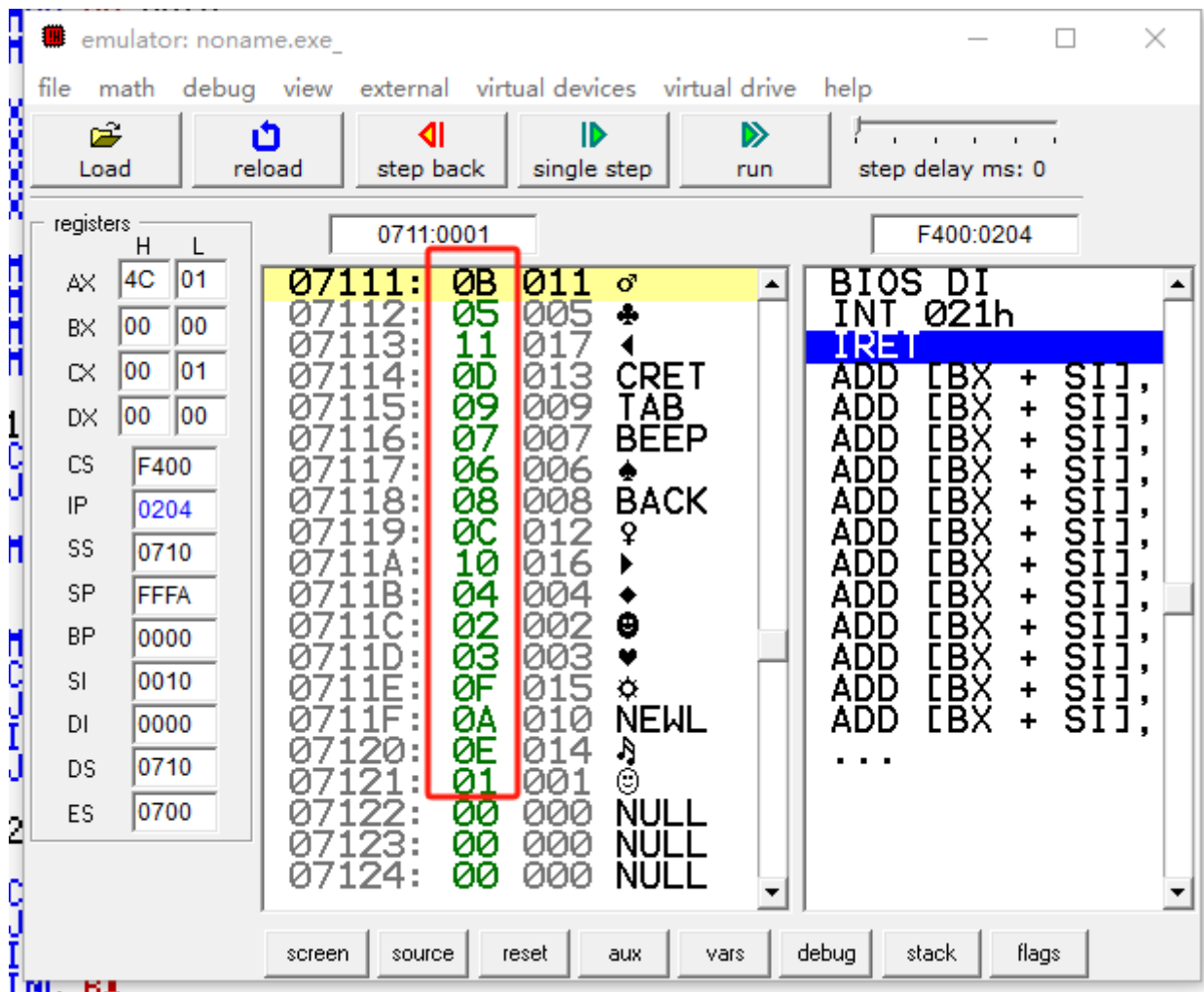
INT 21H

```
CODE ENDS
END START
```

### 三、结果与分析



其中:



在预先创建的VAR2中存储着出列的顺序。

那么依次是11, 5, 17, 13, 9, 7, 6, 8, 12, 16, 4, 2, 3, 15, 10, 14, 1

## 四、收获及心得体会

通过完成这个实验，我学到了如何使用汇编语言模拟实际生活中的问题。掌握了循环队列的使用，了解了在汇编程序中如何进行指针操作，以及如何通过循环结构来模拟实际场景。这次实验增强了我对汇编语言的掌握程度，提高了我在编程中的逻辑思维能力。

同时，我学会了如何将问题拆分为小步骤，逐步实现，这样更容易控制程序的复杂度。此外，通过对问题的分析和思考，我学到了如何将实际问题映射为计算机程序的逻辑结构。这次实验不仅加深了我对汇编语言的理解，也锻炼了我解决问题的能力。