



第11章 动态数据结构的C语言实现

动态数组



哈尔滨工业大学

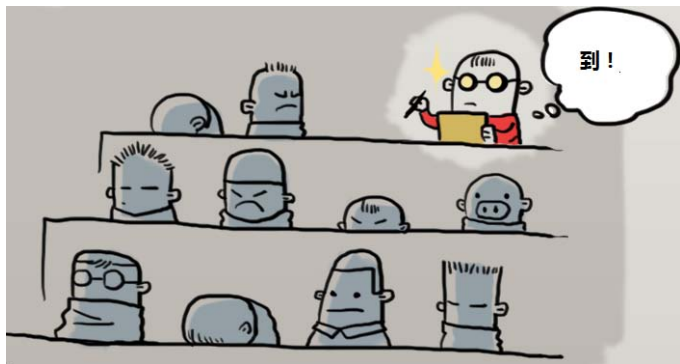
赵玲玲 zhaoll@hit.edu.cn

动态数组

- * 例子：假设我有若干名学生，为了震慑不来上课的同学，欲开发一套自动点名程序，每次课随机抽取若干名学生点名，给出结课后所有学生的出勤分数。

- * 假设：

- * 一共6次课
- * 每个学生出勤满分为5分
- * 如果出现一次缺勤扣1分，两次缺勤扣3分，三次及以上缺勤扣5分

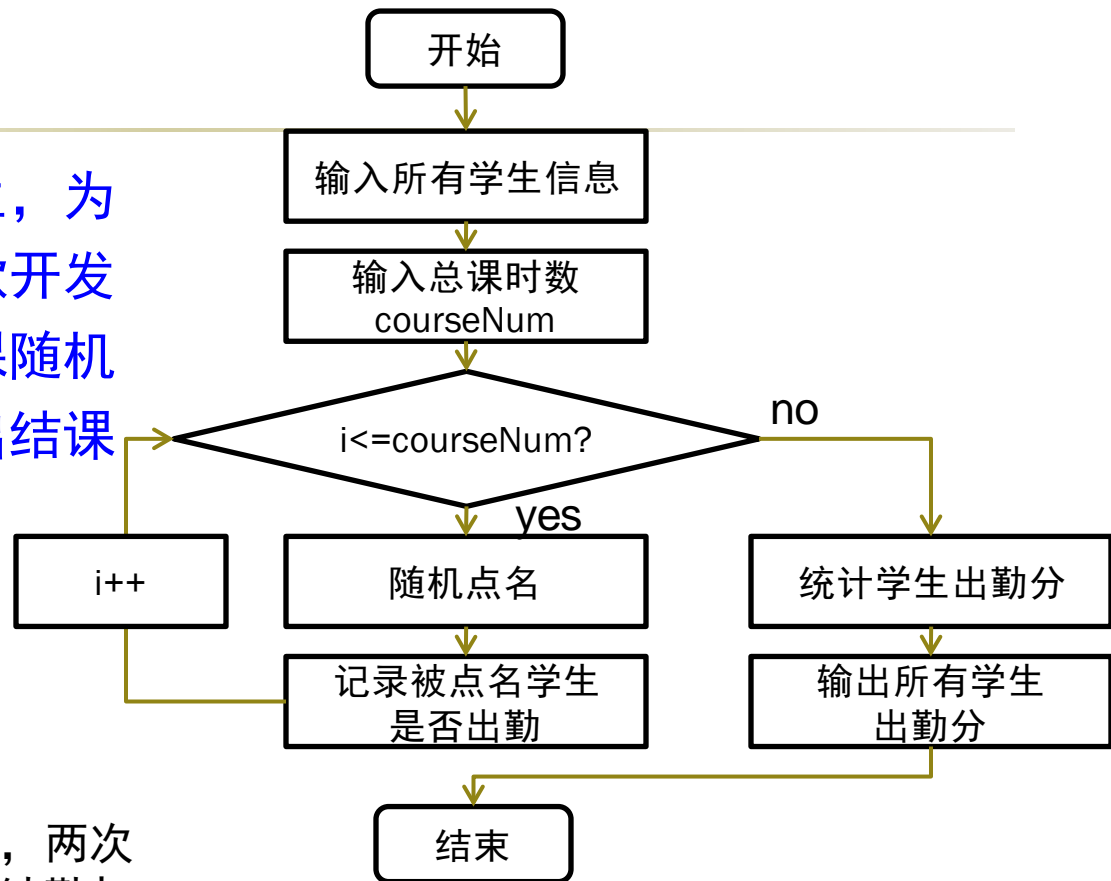


动态数组

- * 例子：假设我有若干个学生，为了震慑不来上课的同学，欲开发一套自动点名程序，每次课随机抽取若干个学生点名，给出结课后所有学生的出勤分数。

* 假设：

- * 一共6次课
- * 每个学生出勤满分为5分
- * 如果出现一次缺勤扣1分，两次缺勤扣3分，三次及以上缺勤扣5分



动态数组

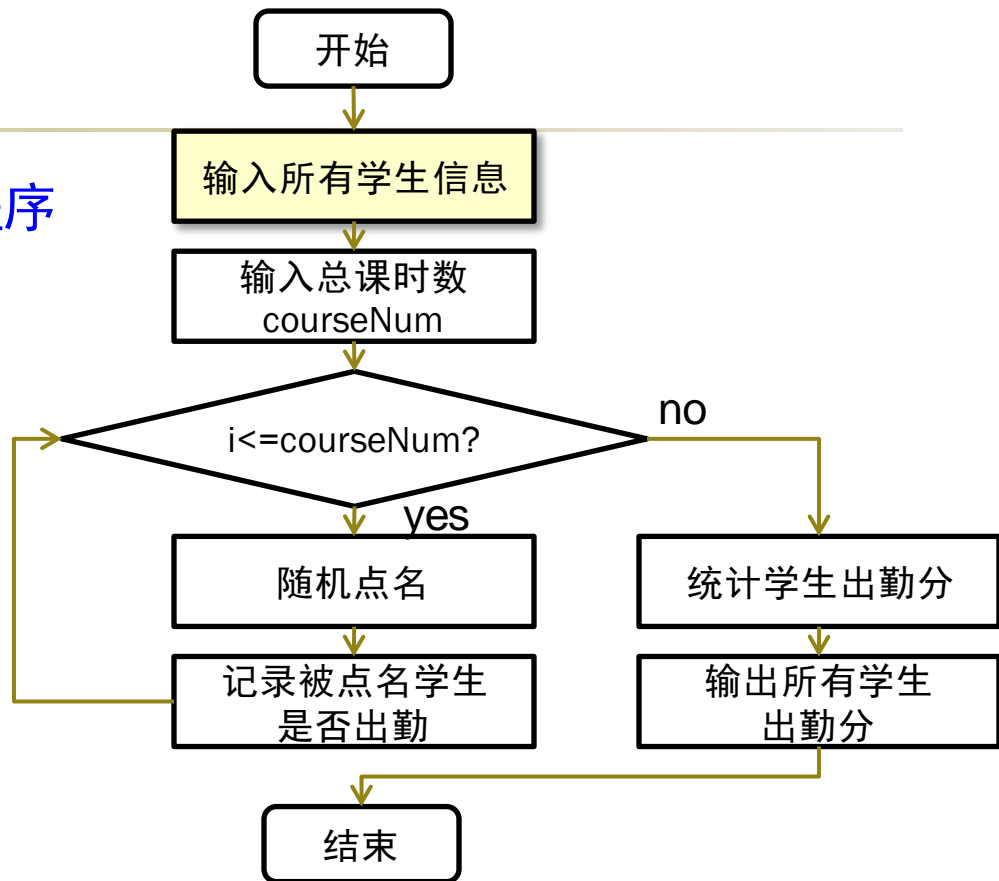
* 例子：开发一套自动随机点名程序

InputInfo()

怎么把学生信息记录下来呢？

数组？

结构体？



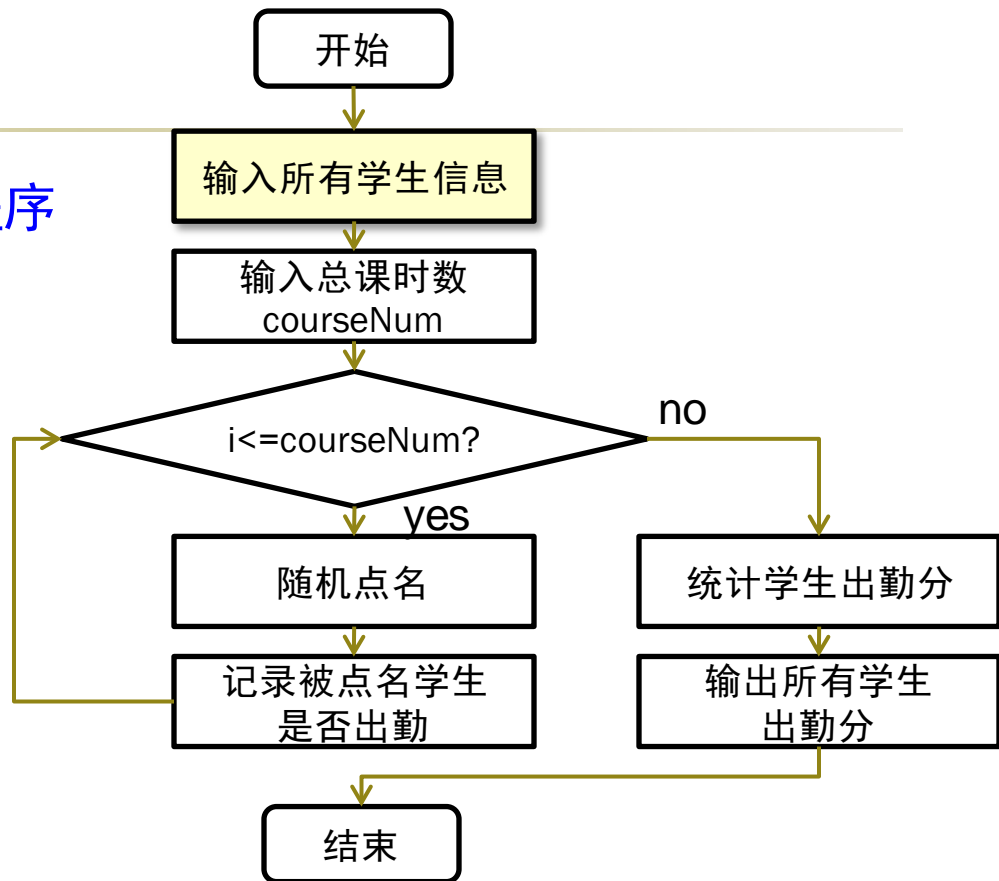
动态数组

* 例子：开发一套自动随机点名程序

数组解法

InputInfo()

```
#define MAX 100
char stuName[MAX][20];
char stuID[MAX][10]
int stuScore[MAX];
Int stuAbsentNum[MAX];
```



动态数组

* 例子：开发一套自动随机点名程序

数组解法

InputInfo()

```
#define MAX 100
char stuName[MAX][20];
char stuID[MAX][10]
int stuScore[MAX];
Int stuAbsentNum[MAX];
```

	stuID	stuName	stuScore	
0	030101	alice	5	0
1	030102	jason	5	0
2	030103	peter	5	0
3	030104	sara	5	0
4	030105	jerry	5	0
...	...			
MAX-1				

stuAbsentNum

动态数组

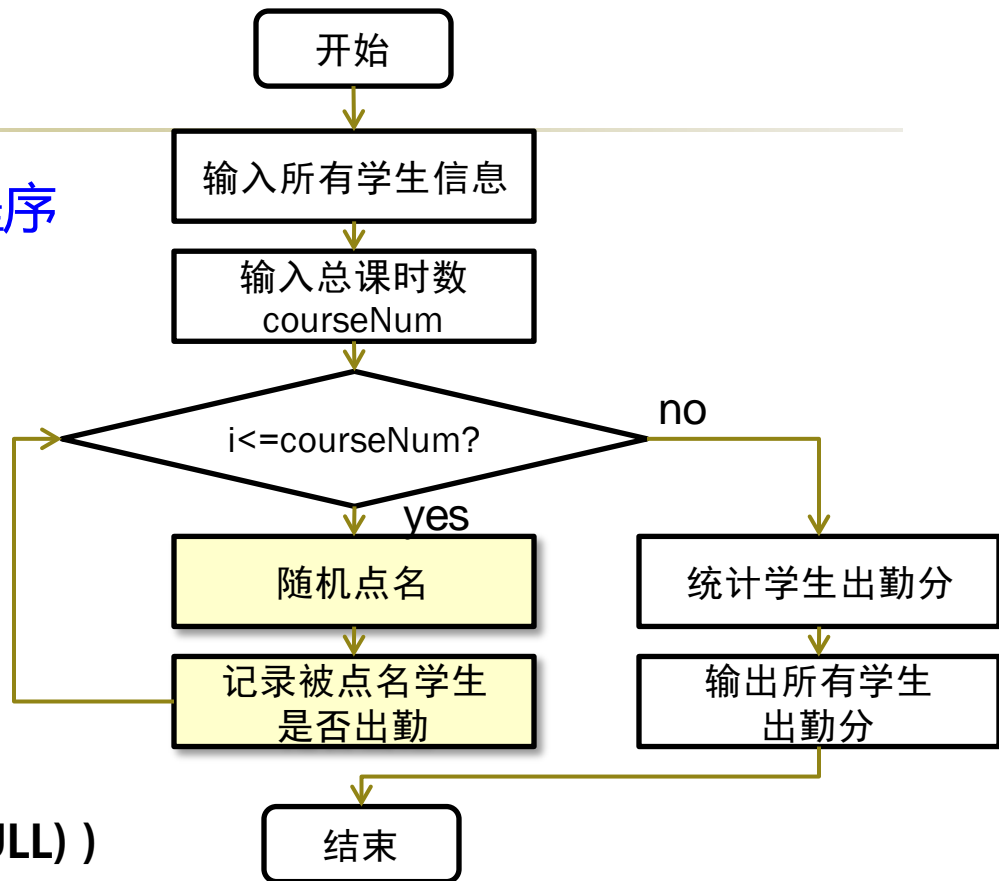
* 例子：开发一套自动随机点名程序

RandomCall()

随机点名什么东东？
被点名的孩子怎么记录？

随机数

rand()+ srand((unsigned int)(time(NULL)))



动态数组

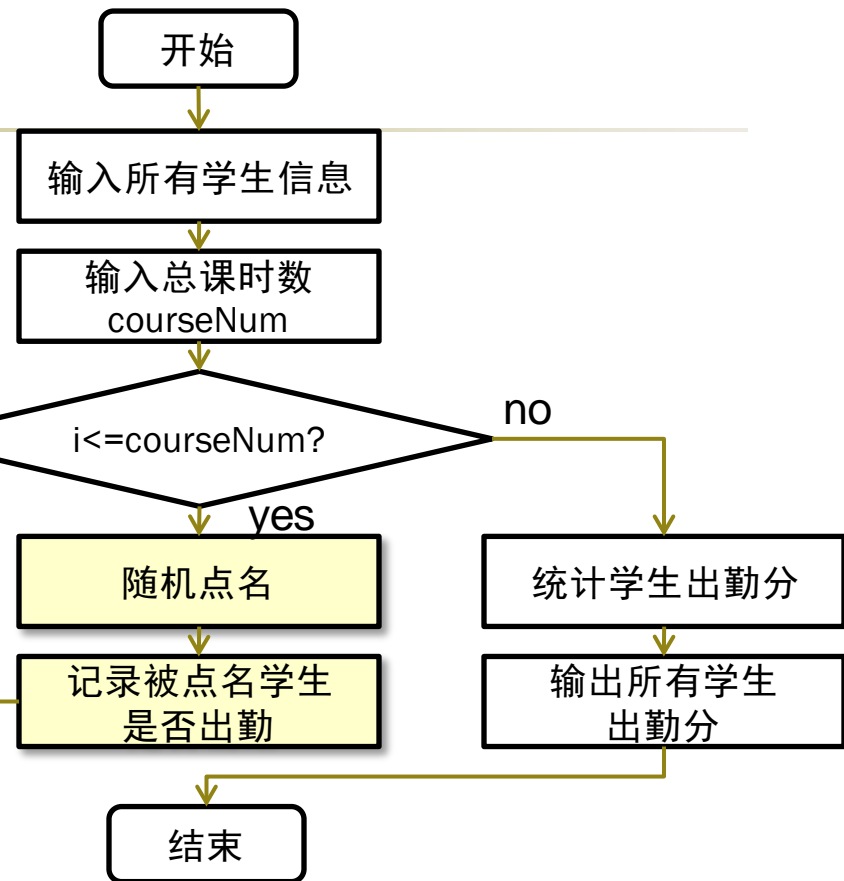
* 例子：开发一套自动随机点名程序

RandomCall()

随机点名什么东东？
被点名的孩子怎么记录？

随机数

新数组记录
点名下标

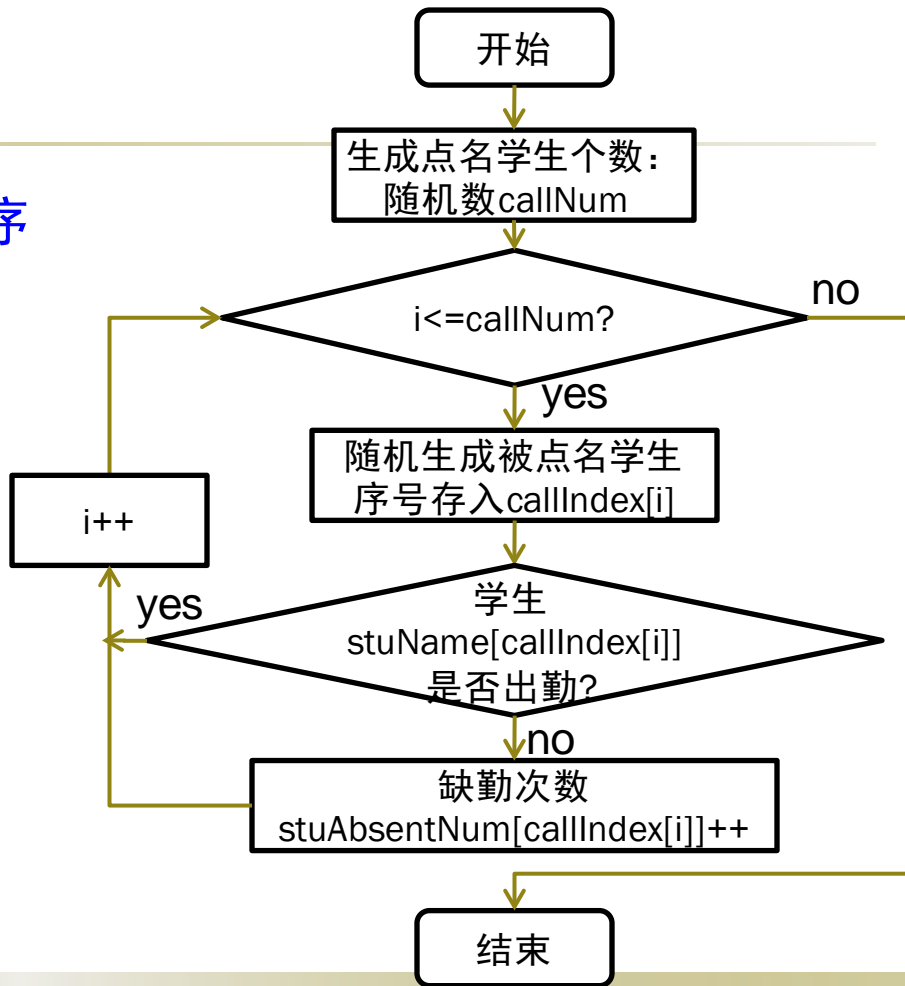
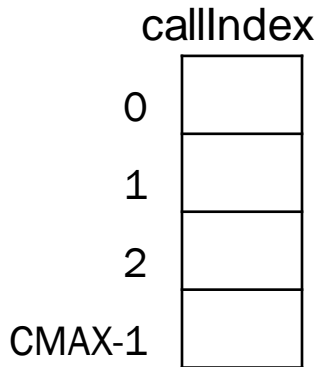


动态数组

* 例子：开发一套自动随机点名程序

RandomCall()

```
#define CMAX 4  
int callIndex[CMAX];
```



动态数组

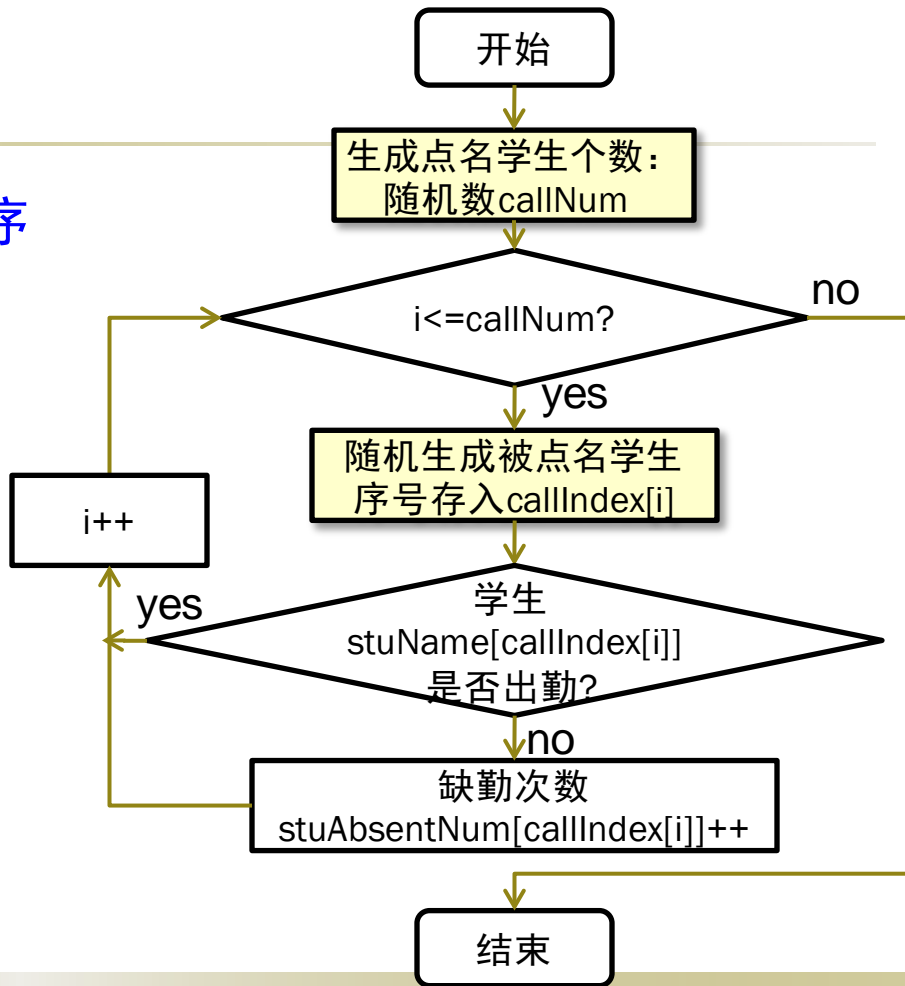
* 例子：开发一套自动随机点名程序

RandomCall()

...

```
srand(time(NULL));  
callNum = rand()%CMAX;  
callIndex[i] = rand()%stuNum;
```

...



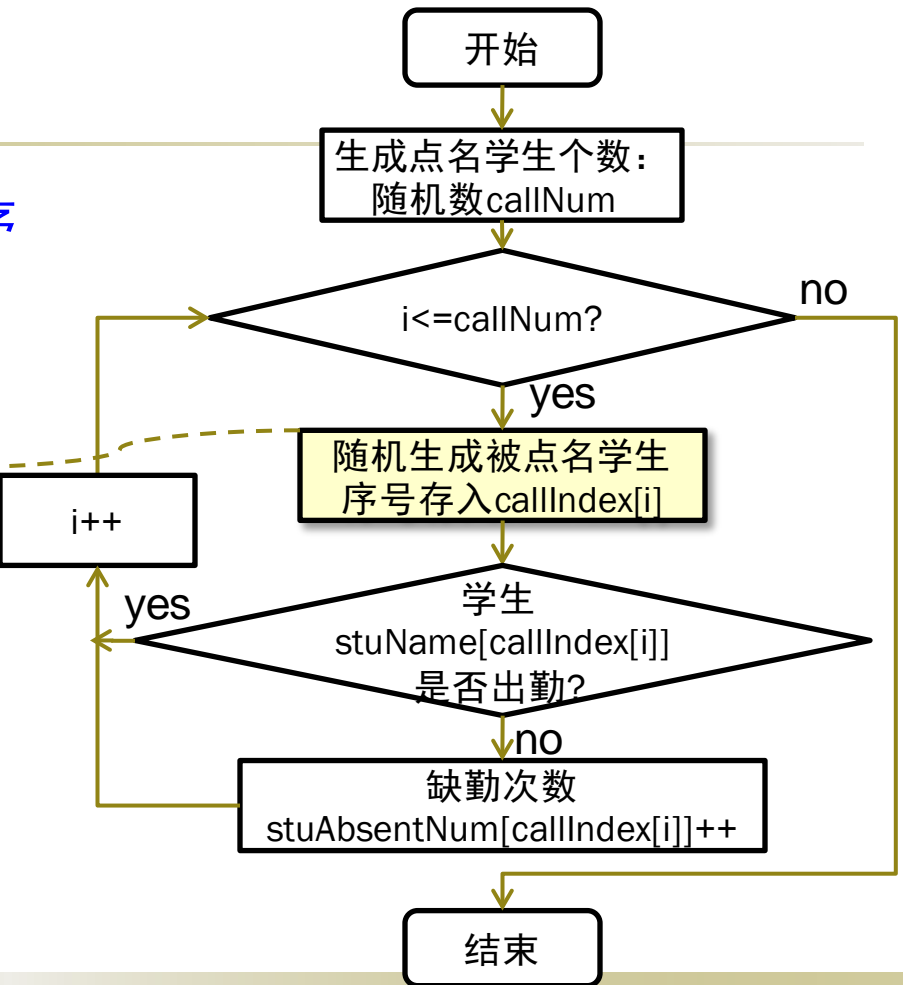
动态数组

* 例子：开发一套自动随机点名程序

RandomCall()

	stuID	stuName
0	030101	alice
1	030102	jason
2	030103	peter
3	030104	sara
4	030105	jerry
...	...	
MAX		

callIndex
3
0
1



动态数组

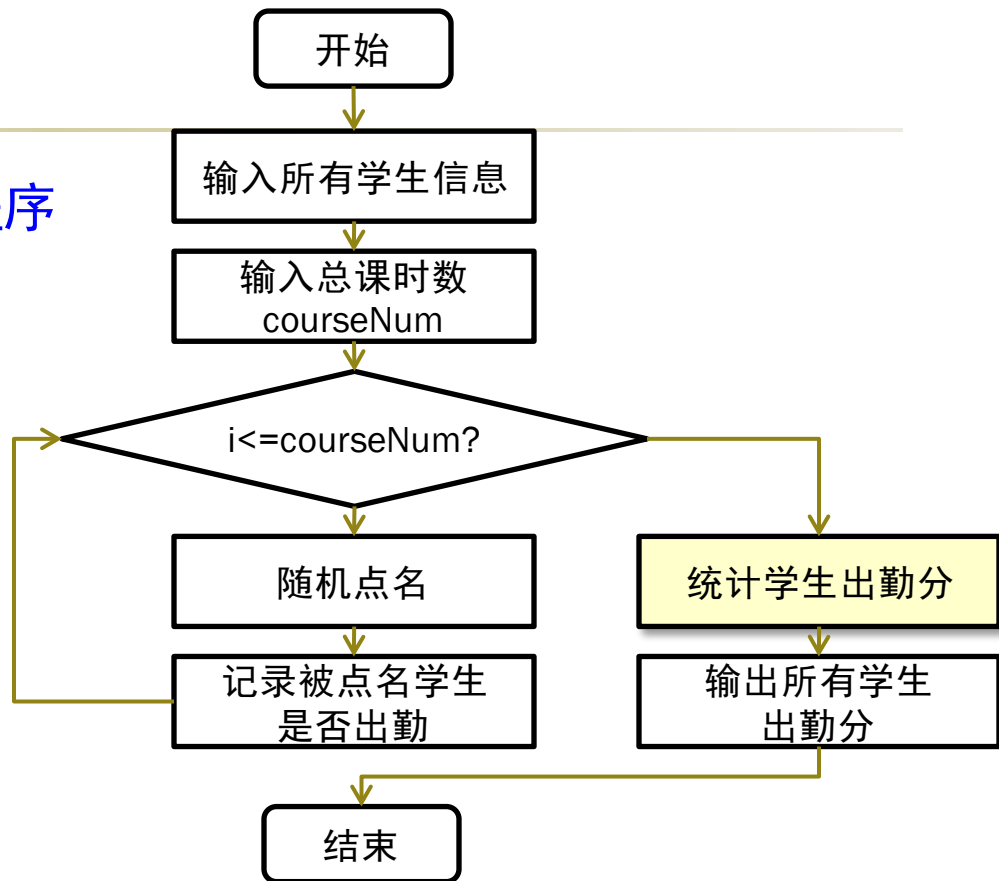
* 例子：开发一套自动随机点名程序

* SumScore()

按照扣分规则计算出
勤分：

2	3
3	5
4	5

switch语句



动态数组

- * 例子：假设我有若干名学生，为了震慑不来上课的同学，欲开发一套自动点名程序，每次课**随机抽取若干个**学生点名，给出结课后所有学生的出勤分数。

静态数组解法

优点：好实现

缺点：
空间浪费
多多

动态数组

* 例子：开发一套自动随机点名程序---动态数组解法

P1 = (char *)malloc(sizeof(char)*stuNum*20);
//存放学生的名字

P2 = (int *)malloc(sizeof(int)*stuNum);
//存放每个学生的出勤分数

P3 = (int *)malloc(sizeof(int)*stuNum);
//存放每个学生的缺勤次数

P4 = (int *)malloc(sizeof(int)*callNum);
//存放每次点名生成的随机数

	stuID	stuName	stuScore	
0	030101	alice		0
1	030102	jason		0
2	030103	peter		0
3	030104	sara		0
stuNum-1	030105	jerry		0

stuAbsentNum

动态数组

- * 例子：假设我有若干名学生，为了震慑不来上课的同学，欲开发一套自动点名程序，每次课随机抽取若干名学生点名，给出结课后所有学生的出勤分数。

优点：空间不浪费，灵活

缺点：操作相对复杂一些，用后回收

动态数组解法1
一维动态数组

动态数组

* 例子：开发一套自动随机点名程序---动态数组解法2

P1 = (char *)calloc(stuNum*20, sizeof(char));
//存放学生的名字

P2 = (int *)calloc(stuNum, sizeof(int));
//存放每个学生的出勤分数

P3 = (int *) calloc(stuNum, sizeof(int));
//存放每个学生的缺勤次数

P4 = (int *) calloc(callNum, sizeof(int));
//存放每次点名生成的随机数

	stuID	stuName	stuScore	
0	030101	alice		0
1	030102	jason		0
2	030103	peter		0
3	030104	sara		0
stuNum-1	030105	jerry		0

stuAbsentNum

二维动态数组

- * 例子：假设我有若干名学生，为了震慑不来上课的同学，欲开发一套自动点名程序，每次课**随机抽取若干**名学生点名，给出结课后所有学生的出勤分数。

优点：空间不浪费，更像数组

缺点：操作相对复杂一些，用后回收

动态数组解法2
二维动态数组

动态数组-扩展问题

- * 例子：假设我有若干名学生，为了震慑不来上课的同学，欲开发一套自动点名程序，每次课随机抽取若干名学生点名，给出结课后所有学生的出勤分数。

- * 万能点名程序：

- * 上课次数可变
- * 每次课可以点名也可以不点名
- * 要求所有学生都至少被点名过一次
- * 出勤分的规则可以改变，比如出现一次缺勤扣1分，两次缺勤扣2分，三次扣4分，...
- *

