1：calloc

功 能: 在内存的动态存储区中分配n个长度为size的连续空间，函数返回一个指向分配起始地址的指针；如果分配不成功，返回NULL。

跟malloc的区别:

calloc在动态分配完内存后，自动初始化该内存空间为零，而malloc不初始化，里边数据是随机的垃圾数据。

用 法:

void *calloc(unsigned n,unsigned size);

头文件: stdlib.h或malloc.h

相关函数: malloc、realloc、free

例子:

```
#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    int i;
    int *pn=calloc(10,sizeof(int));
    for(i=0;i<10;i++)
    printf("%3d",*pn++);
    printf("/n");
    free(pn);
    return 0;
}
```

输出十个0。

2：malloc

功 能: 在内存的动态存储区中分配n个长度为size的连续空间，函数返回一个指向分配起始地址的指针；如果分配不成功，返回NULL。

用法: void* malloc(unsigned size);

头文件: stdlib.h

例子:

```
char* p;
p=(char*)malloc(20);
```

3：relloc

功 能:

realloc是给一个已经分配了地址的指针重新分配空间,参数ptr为原有的空间地址,newsize是重新申请的地址长度

头文件：stdlib.h

用法:

void* calloc(size_t numElements, size_t sizeOfElement);

例子:

char* p;

p=(char*)malloc(sizeof(char)*20);

p=(char*)realloc(p,sizeof(char)*40);

tip：realloc调用形式为(类型*)realloc(*ptr，size)：将ptr内存大小增大到size。

4：free

功 能:

释放ptr所指向的一块内存空间

用法:

free(void*ptr);

头文件：stdlib.h