

指针

存储变量的地址 指针只能存地址!!!

定义

1. type *name;

```
char *p = NULL; //变量p 类型char*  
//如果不确定p的指向 暂时赋值为 NULL(空)
```

类型

1. 步长、取值大小

```
printf("%p\n", p+1); //char *指针+1 ---> 加sizeof(char)个字节
```

2. 存地址---》变量大小8字节

动态开辟

1. malloc

```
void *malloc(size_t size);  
//动态开辟size个字节的内存空间。
```

2. free

```
void free(void *ptr);  
//释放所开辟的动态内存空间。  
//谁申请谁释放
```

3. calloc

```
void *calloc(size_t nmemb, size_t size);  
//在内存的动态存储区中开辟nmemb个长度为size的连续空间。  
//和malloc不同的是calloc会自动初始化内存空间为零,  
//而malloc不做初始化, 分配的内存空间为随机值。
```

```
int *ptr = malloc(sizeof(int)); //分配空间
memset(ptr,0,sizeof(int)); //初始化
//上面两句等价于calloc.
```

4. realloc

```
void *realloc(void *ptr, size_t size);
// (数据类型) realloc (要改变内存大小的指针名, 新的内存大小)
// 新的内存大小可以比原内存大小大, 也可以比原内存大小小,
// 如果比原内存大小大, 则新分配部分不会被初始化,
// 如果比原内存要小, 可能会丢失数据。
```

5. 选择

1. 只申请一块内存空间且不需要初始化的时候用 `malloc` ;
2. 申请一块内存空间需要初始化的时候用 `calloc` ;
3. 申请一块内存空间后需要改变内存空间大小用 `realloc` ;