

第二节课内容总结

影响对象内存的因素

对象里面存储了一个isa指针 + 成员变量的值，isa指针是固定的，占8个字节，所以影响对象内存的只有成员变量（属性会自动生成带下划线的成员变量）

对象的内存分布

在对象的内部是以8字节进行对齐的。

苹果会自动重成员变量的顺序，将占用不足8字节的成员挨在一起，凑满8字节，以达到优化内存的目的。

联合体（union）

联合体又叫共用体，union就是在内存中划了一个足够用的空间，至于你怎么玩~它不管~！联合体的成员变量就相当于为这块内存空间开辟了几个访问途径，他们共享这一块内存。

联合体的大小计算奉行俩个规则

- 1.联合体大小必须能容纳联合体中最大的成员变量
- 2.通过1计算出的联合体大小必须是联合体中占内存大小最大的基本数据类型大小的整数倍

联合体和结构体的区别

结构体(struct)中所有变量是“共存”的，而联合体(union)中是各变量是“互斥”的，只能存在一个。struct内存空间的分配是粗放的，不管用不用，全部分配。这样带来的一个坏处就是对于内存的消耗要大一些。但是结构体里面的数据是完整的。

联合体里面的数据只能存在一个，但优点是内存使用更为精细灵活，也节省了内存空间。

位域

位域的宽度不能超过前面数据类型的最大长度，比如int占4个字节也就是32位，那后面的数字就不能超过32。

一个位域存储在同一个字节中，如一个字节所剩空间不够存放另一位域时，则会从下一单元起存放该位域。

位域能够节省一定的内存空间。

nonPointerIsa

nonPointerIsa是内存优化的一种手段。isa是一个Class类型的结构体指针，占8个字节，主要是用来存内存地址的。但是8个字节意味着它就有 $8*8=64$ 位。存储地址根本不需要这么多的内存空间。而且每个对象都有个isa指针，这样就浪费了内存。所以苹果就把和对象一些息息相关的东西，存在了这块内存空间里面。这种isa指针就叫nonPointerIsa。

nonPointerIsa存储的内容有：

isa指针的数据结构

arm64 (模拟器)

```
uintptr_t nonpointer      : 1;
uintptr_t has_assoc       : 1;
uintptr_t weakly_referenced : 1;
uintptr_t shiftcls_and_sig : 52;
uintptr_t has_sidetable_rc : 1;
uintptr_t extra_rc        : 8
```

arm64 (真机)

```
uintptr_t nonpointer      : 1;
uintptr_t has_assoc       : 1;
uintptr_t has_cxx_dtor    : 1;
uintptr_t shiftcls        : 33;
uintptr_t magic           : 6;
uintptr_t weakly_referenced : 1;
uintptr_t unused          : 1;
uintptr_t has_sidetable_rc : 1;
uintptr_t extra_rc        : 19
```

x86_64

```
uintptr_t nonpointer      : 1;
uintptr_t has_assoc       : 1;
uintptr_t has_cxx_dtor    : 1;
uintptr_t shiftcls        : 44;
uintptr_t magic           : 6;
uintptr_t weakly_referenced : 1;
uintptr_t unused          : 1;
```

```
uintptr_t has_sidetable_rc : 1;  
uintptr_t extra_rc          : 8
```