



Structures

- ① Structures
- ② Data Alignment

Structures

r in %rdi i in %rsi

```
struct rec {
    int i;
    int j;
    int a[2];
    int *p;
};
```

```
(%rdi), %eax
movl
         %eax,4(%rdi)
movl
         8(%rdi,%rsi,4),%rax
leaq
         4(%rdi),%eax
movl
addl
          (%rdi),%eax
cltq
         8(%rdi,%rax,4),%rax
leaq
         %rax,16(%rdi)
movq
 8
             16
                         24
```

i j a[0] a[1] p

```
void sp init(struct prob *sp) {
Example
                 sp->s.x
                          = sp->s.y;
                 sp->p = &(sp->s.x);
                 sp->next =
                           Sp
struct prob {
  int *p;
  struct {
                       sp init:
     int x;
                                12(%rdi),%eax
                          movl
     int y;
                                %eax,8(%rdi)
                          movl
  } s;
                                8(%rdi),%rax
                          leaq
  struct prob *next;
                                %rax,(%rdi)
                          movq
};
                                %rdi,16(%rdi)
                          movq
                          ret
                 8
                            16
                      12
                                       24
```

S.X

S.y

next

prob

Example

long fun(struct ELE *ptr)

```
fun:
  movl $0,%eax
       .L2
  jmp
.L3:
  addq (%rdi),%rax
  movq 8(%rdi),%rdi
.L2:
  testq %rdi,%rdi
  jne
       .L3
  rep; ret
```

```
long ret=0;
while(ptr) {
    ret += ptr->v;
    ptr = ptr->p;
}
return ret;
```

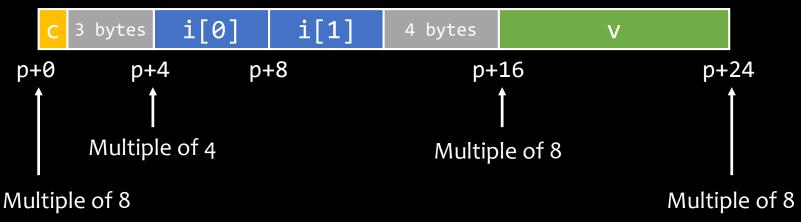
```
struct ELE {
   long v;
   struct ELE *p;
};
```

Data Alignment

```
c i[0] i[1] v
p p+1 p+5 p+9 p+17
```

- 1 Primitive data type requires K bytes
- ② Address must be multiple of K

padding



struct S1 {

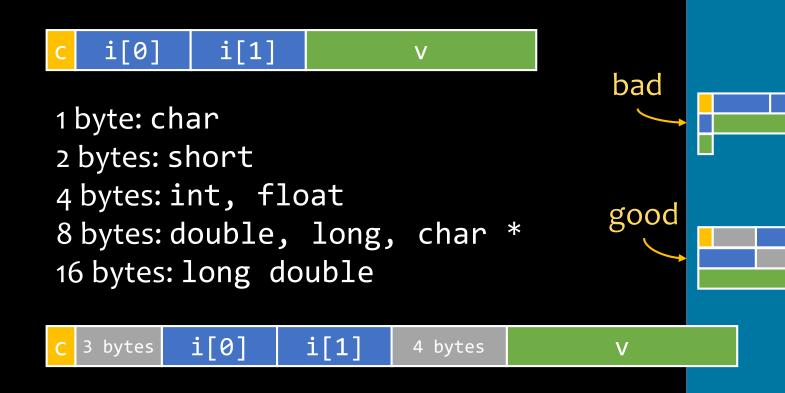
*p;

char c;

int i[2];

double v;

Why?



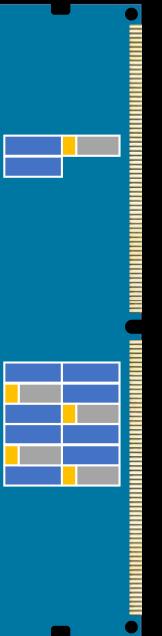
Example

```
struct S1 {
   int i;
   char c;
   int j;
};
```

- ③ Overall structure must be multiple of K
- Put large data types first

```
struct S2 {
   int i;
   int j;
   char c;
};
```

struct S2 d[4];



Saving space

```
struct S4 {
                   struct S4 d[8];
  char c;
  int i;

    Put large data types first

  char d;
} *p;
struct S5 {
  int i;
  char c;
  char d;
} *p;
                   struct S5 d[8];
```

```
16
                                        24
0
  Example
  struct P1{ int i; char c; int j; char d; };
  struct P2{ int i; char c; char d; long j; };
      c d 2
  struct P3{ short w[3]; char c[3]; };
   W
  struct P4{ short w[5]; char *c[3]; };
                    6
       W
  struct P5{ struct P3 a[2]; struct P2 t };
      a[1]
                       a[2]
```

Summary

- Structures
- Data Alignment



Eric Steven Raymond

Author of "The Cathedral and the Bazaar"

