Pointers

Pointer Arithmetic

Pointer Arithmetic

- Since a pointer is just a memory address, we can add an integer to it to traverse an array.

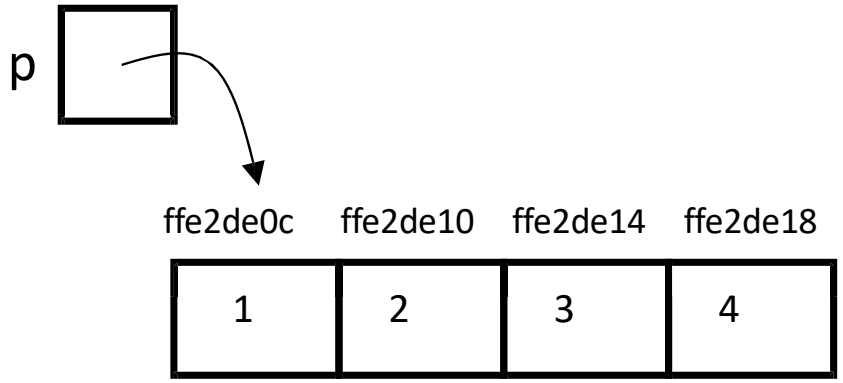
- If p is a pointer to the first element in an array, p+1 returns a pointer to the next array element.

p
= *1;

- p+1 doesn't add 1 to the memory address, it adds the size of the array element.

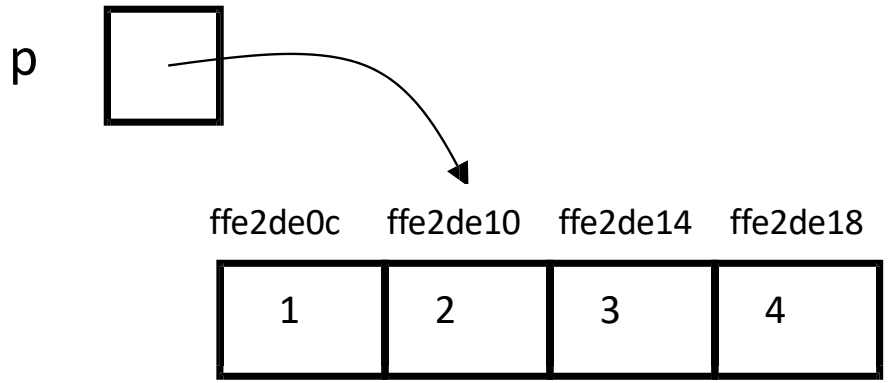
Pointer Arithmetic

```
int list[] = {1, 2, 3, 4};
int *p = list;    /* same as p = &list[0] */
printf("%x",p);   /* prints ffe2de0c */
```



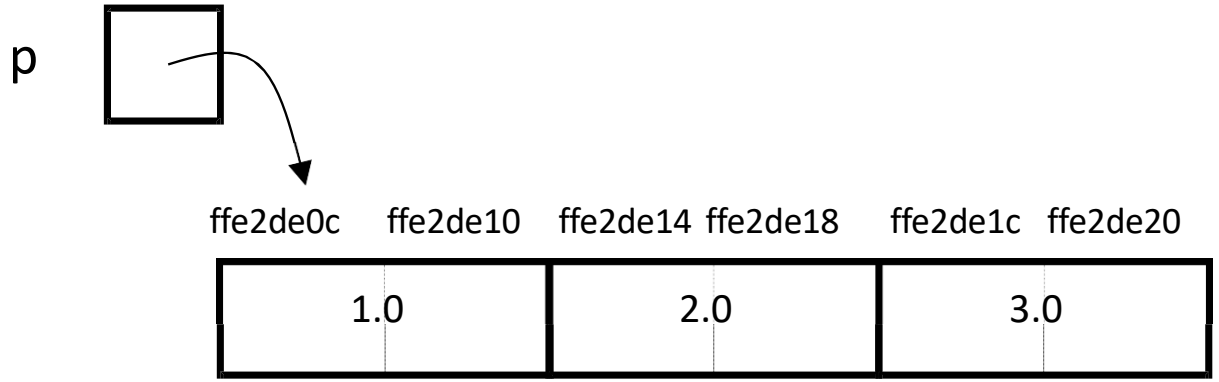
Pointer Arithmetic

```
int list[] = {1, 2, 3, 4};
int *p = list;     /* same as p = &list[0] */
printf("%x",p);    /* prints ffe2de0c */
p = p + 1;         /* p increases by 4 */
printf("%x",p);    /* prints ffe2de10 */
```



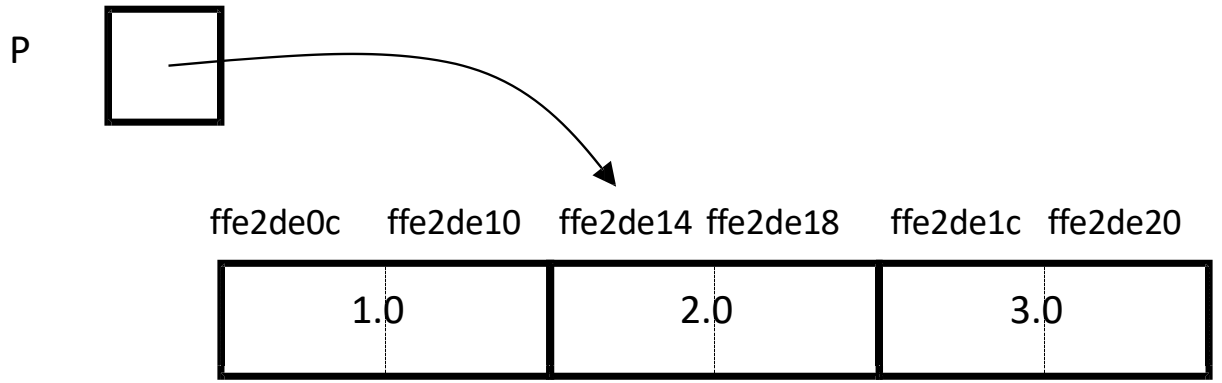
Pointer Arithmetic

```
double list2[] = {1.0, 2.0, 3.0};
double *p = list2;    /* same as p = &list2[0] */
printf("%x", p);      /* prints ffe2de0c */
```



Pointer Arithmetic

```
double list2[] = {1.0, 2.0, 3.0};
double *p = list2;   /* same as p = &list2[0] */
printf("%x",p);      /* prints ffe2de0c */
p = p + 1;           /* P increases by 8 bytes */
printf("%x",p);      /* prints ffe2de14 */
```



Pointer Arithmetic

- So what's valid pointer arithmetic?
  - Add an integer to a pointer.
  - Subtract 2 pointers (in the same array).
  - Compare pointers (<, <=, ==, !=, >, >=)
  - Compare pointer to NULL (indicates that the pointer points to nothing).
- Everything else is illegal since it makes no sense:
  - adding two pointers
  - multiplying pointers
  - subtract pointer from integer

Pointer Arithmetic Examples

- If two pointers point to elements of a same array, then there are notions of subtraction and comparisons between the two pointers.

```
int a[10], *p, *q , i;
p = &a[2];
q = &a[5];

i = q - p;        /* i is 3*/
i = p - q;        /* i is -3 */
a[2] = a[5] = 0;
i = *p - *q;      /* i = a[2] – a[5]  */

p < q;            /* true */
p == q;           /* false */
p != q;           /* true */
```

Homework

1. Write a C function that uses pointer arithmetic to traverse the array and copy the contents of one array to another. The prototype of the function is as follows:

   void copy(int *from, int *to, int size);

2. If p is a pointer to the first element of an array, explain what is the difference between x = *p++ and x = (*p)++.

3. How many of the following are invalid?
   a) pointer + integer
   b) integer + pointer
   c) pointer + pointer
   d) pointer – integer
   e) integer – pointer
   f) pointer – pointer
   g) compare pointer to pointer
   h) compare pointer to integer
   i) compare pointer to 0
   j) compare pointer to NULL