

# Pointers

Yung-Hsiang Lu

# Computer Memory

The Dell Online Store: Build Your System - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

http://configure.us.dell.com/dellstore/config.aspx?oc=bwdwjap&c=us&l=en&s=bsd&cs=04

United States Buy Online or Call 1-800-303-0729

Shop Support Community Keyword Search

Laptops & Netbooks Desktops & Workstations Servers, Storage & Networking Printers, Ink & Toner Electronics & Accessories Software Services Solutions

Small & Medium Business Dell recommends Windows V

Need help making a selection? View our comprehensive Help Me Choose center.

intel

LIVE CHAT CLICK TO CHAT

1 Customize 2 Accessories 3 Recommended Add-ons & Services 4 Review & Add to Cart

SWITCH TO LIST VIEW

**SELECT MY MEMORY** ←

- 4GB, DDR2 SDRAM FBD Memory, 800MHz, ECC (4 DIMMS) [add \$480]
- 2GB, DDR2 SDRAM FBD Memory, 800MHz, ECC (2 DIMMS) [add \$10]
- 4GB, DDR2 SDRAM FBD Memory, 667MHz, ECC (4 DIMMS) [add \$460]
- 2GB, DDR2 SDRAM FBD Memory, 667MHz, ECC (2 DIMMS) [Included in Price]

Precision T7

Starting Price .....  
Instant Savings .....

**Subtotal** .....

Obtain enough storage and performance for your most demanding applications and lease a Larger Hard Drive. Upgrade to a 500GB Hard Drive Now!

Lease from \$92/mo. (48 prnts)

Done

# How is Memory Used?

In a computer, memory is accessed using **addresses**.

Address	Data
0X00000000	...
...	...
0x08001F00	'a'
...	...
0X1A0088F0	642
...	...

# How to Find the Address?

```
int x = 123;
```

```
&x /* ampersand */           ⇒ address of x
```

Address	Data
somewhere (&x)	123

A program **cannot** control **where** a variable (such as x) is placed. It is determined by the compiler + run time system. A program can control the value at this address.

```
x = 654;      /* change the value */
```

Address	Data
somewhere (&x)	654

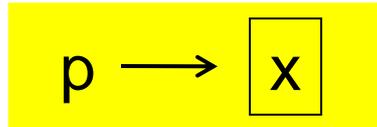
```
&x = 90;     /* error, cannot change x's address */
```

# Pointer

```
int x = 123;
```

```
int * p;    /* p is a pointer to an integer */
```

```
p = &x;    /* p points to x */
```



& x        ⇒ x's address

\*p        ⇒ value at the address pointed by p; it is 123

Address	Data
somewhere (&x)	123
...	...
somewhere (&p)	&x

A red arrow originates from the '&x' data in the 'somewhere (&p)' row and points to the 'somewhere (&x)' row, indicating that the pointer variable 'p' holds the address of 'x'.

# L value and R value

- L value means the left side of an assignment.
- R value means the right side of an assignment.

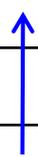
```
int x = 123; /* x is L value, 123 is R value */
```

```
int y = 987; /* y is L value */
```

```
x = y; /* x is L value; y is R value */
```

- This means
  - take the **address** of x (somewhere, we cannot control)
  - take the **value** of the address of y
  - assign the value to the address
  - result: x's value is 987.

Address	Data
somewhere (&x)	123
...	...
somewhere (&y)	987



# Change Values through Pointers

```
int x = 123;  
int * p = & x; /* p points to x */
```

Address	Data
somewhere (&x)	123
...	...
somewhere (&p)	& x



```
*p = 456;  
/* take the value of p (x's address) as the address and  
assign 456 to this address; x becomes 456 */
```

Address	Data
somewhere (&x)	456
...	...
somewhere (&p)	& x



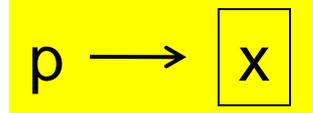
# Pointer Operation

```
int * p;          /* p is a pointer to an integer */
```

```
int x = 123;
```

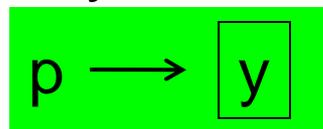
```
int y = 987;
```

```
p = &x;          /* p points to x */
```



```
*p = 456;        /* x is 456 now */
```

```
p = &y;          /* p points to y */
```



```
*p = -765;       /* y is -765 now */
```

Address	Data
somewhere (&x)	456
...	...
somewhere (&p)	&x
somewhere (&y)	987

Address	Data
somewhere (&x)	456
...	...
somewhere (&p)	&y
somewhere (&y)	-765

# Common Mistake

A program cannot control addresses.

```
65 = 123;
```

```
/* cannot assign value 123 to the address 65 */
```

```
int * p = 86;
```

```
/* warning message, the
```

```
address is likely occupied by another program. */
```

Address	Data
86	...
somewhere (&p)	86



```
double d = 0.84;
```

```
int * p = & d;      /* warning message */
```

Which is correct? You may choose multiple answers.

```
int x = 6543;  
int * p = & x;
```

- A) The value of p is 6543.
- B) The address of p is 6543.
- C) The address of x is 6543.
- D) \*p = 123 makes x's value 123.
- E) p = 123 makes x's value 123.
- F) & p = 123 makes x's value 123.

Correct - Click anywhere to  
continue

Incorrect - Click anywhere to  
continue

Your answer:

You did not answer this question

You must answer the question  
before continuing

Submit

Clear

Which is correct? You may choose multiple answers.

```
int x = 123;  
int y = 567;  
int * p = & x;
```

- A)  $p = y$ ; makes  $x$ 's value 567.
- B)  $p = \& y$ ; does not change  $x$ 's value nor  $y$ 's value.
- C)  $p = y$ ; makes  $x$ 's value 567.
- D)  $y = x$ ; makes  $x$ 's value 567.
- E)  $y = \& p$ ; makes  $y$ 's value 123.

Correct - Click anywhere to continue

Incorrect - Click anywhere to continue

Your answer:

You did not answer this question

You must answer the question before continuing

Submit

Clear

# Pointer

<b>Your Score</b>	{score}
<b>Max Score</b>	{max-score}
<b>Number of Quiz Attempts</b>	{total-attempts}

Question Feedback/Review Information Will Appear Here

Continue

Review Quiz