

3/20/19/1

Exam #1 4/1

K

35

 28C4
(int)

location ← Scalar
type
value
name

array

X	
[0]	5
[1]	39
[2]	1
[3]	72
[4]	80

int x[5];

x[3] = 72;

x[2] = x[3] - 71;

x[4] = x[2] + x[3] + 7;

x[0] = 5;

x[x[2]] = 39;

```
int k = 35;
```

```
int x[5] = {5, 39, 1, 72, 80};
```

```
int a[] = {50, 100, 239, 4};
```

```
int b[5] = {6};
```

x

5
39
1
72
80

a	
[0]	50
	100
	239
[3]	4

b	
[0]	6
	0
	0
	0
[4]	0

```
int c[3] = {1, 2, 40, 50}; // ERROR
```

```
for (i=0; i<5; i++)
```

```
    x[i] = 0;
```

```
double s[401] ← #define PI 3.14159---
```

```
for (i=0; i<=360; i++)
```

```
    s[i] = sin ( i * PI / 180.0 );
```

```
// write code to input some number  
// of test scores. Prompt user to #test first
```

```
int n; // # of tests
```

```
int t[100]; // max of 100 tests
```

```
printf("Enter number of tests: ");
```

```
scanf("%d", &n);
```

```
for (i=0; i < n ; i++)
```

```
{
```

```
    printf("Enter test %d: ", i+1);
```

```
    scanf("%d", &t[i]);
```

```
}
```

0	80
1	75
2	90
3	100
7	65
5	88
	?
	?
99	?

// Find biggest
 // where n contains
 // the number of used elements

big = 0; // biggest so far

for (i = 0; i < n; i++)

{
 if (x[i] > big)

big = x[i];

}

// big contains biggest

better way

big = x[0];

for (i = 1; i < n; i++)

{
 if (x[i] > big)

big = x[i];

}

t	
0	80
1	75
2	90
3	100
4	65
5	88
	⋮

```
sum = 0;
for (i = 0; i < n; i++)
    sum = sum + t[i];
```

```
avg = sum / n;
```

```
avg = (double) sum / (double) n;
```

// types ↖ type cast

i int

t[] ~~int~~

sum int

avg float or double

<u>sum</u>	<u>i</u>
0	0
80	1
155	2
245	3