

```

Ex. int i;
      int list[10] = {2, 4, 6, 8, 10, 12, 14, 16, 18, 20};
    
```

```

      i = 0;
      while (i < 10) {
          cout << list[i] << " ";
          i = i + 1;
      }
      cout << endl;
    
```

memory:

	0	1	2	3	4	5	6	7	8	9
list	2	4	6	8	10	12	14	16	18	20

i [0]

output:	2	4	...	20
%				

Try

```
cout << list[10] << endl;
```

```
list[10] = 50;
```

Ex.

```
int i;
```

```
int list[10] = {2, 4, ..., 20};
```

```
i = 0;
```

```
do {
```

```
    cout << list[i] << " ";
```

```
    i = i + 1;
```

```
} while (i < 10);
```

```
cout << endl;
```

```

for (init LCV; test LRC; increment LCV) {
    !
}

```

Ex.

```

int i, list[10] = {2, 4, 6, ... 20};

```

```

for (i=0; i<10; i=i+1) {
    cout << list[i] << " ";
}

```

```

cout << endl;

```


Auto Increment &

Auto Decrement ops.

Shortcuts

	<u>Postfix</u>	<u>Prefix</u>
$a += 1;$	$a++;$	$++a;$

$a -= 1;$	$a--;$	$--a;$
-----------	--------	--------

Ex. for (i=0; i<10; i++) {
:
}

Difference between
Prefix & Postfix :

Ex. `int a=1, b;`
 `b = a++;`
 `cout << a << endl;`
 `cout << b << endl;`

output

```

2
1

```

Ex. `int a=1, b;`
 `b = ++a;`
 `cout << a << endl;`
 `cout << b << endl;`

output

```

2
2

```

Note on lab 5: 2 ways to
count comparisons.

7

① int count = 0;

⋮

while (j >= 1 && list[j] < list[j-1]) {

count++;

⋮

← counts true instances

}

if (j >= 1) {

count++;

← counts possible
false instance

}

Another way

8

```
② int count = 0;  
  ⋮
```

```
  while (i >= 1 && ++count && list[i] < list[i-1]) {  
    ⋮  
  }
```

counts all comparisons.