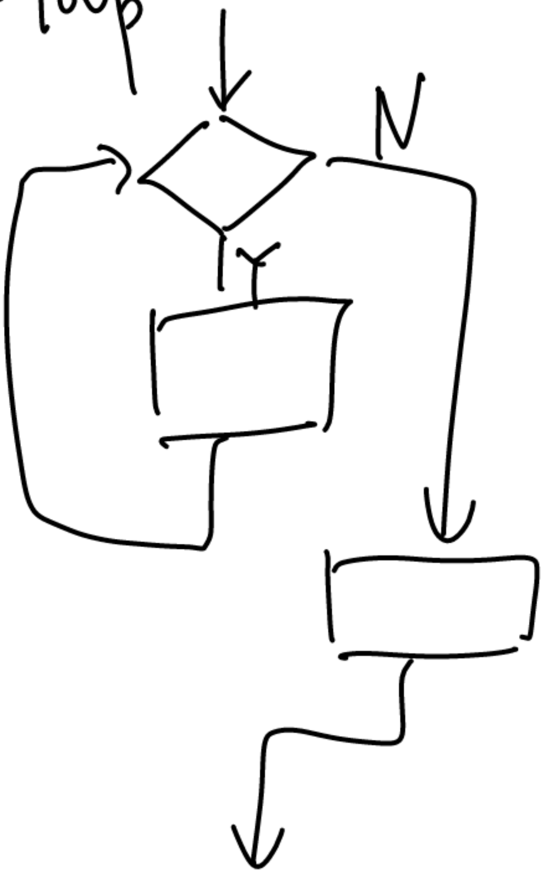
* loop

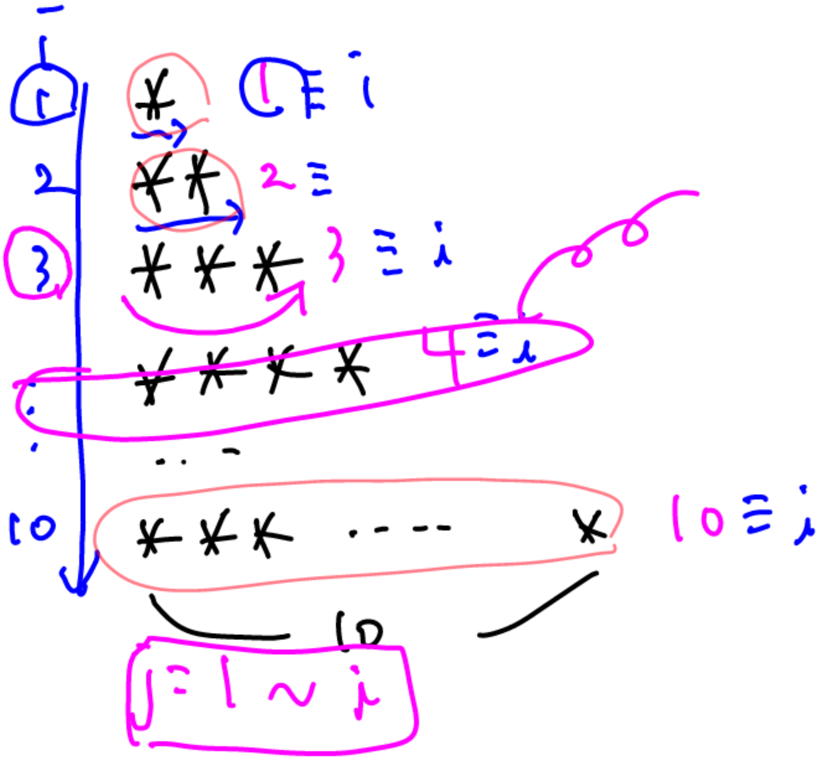


- for -statement
- while - statement
- do ... while statement

```
            ①             ②              ④
for ( statement1; expression1; expression2)
       statement2; ③
```

```
          ①
while ( expression )
     statement; ②
```

```
do        ①
   statement;
while ( expression );
            ②
```



* switch

```
switch ( expression1)
{ [case   expression2 :
        statement1;] [...]
  default :
        statement2;
      [break;]
}
```

ex) 'a'   "abc"

'h'   "hello"

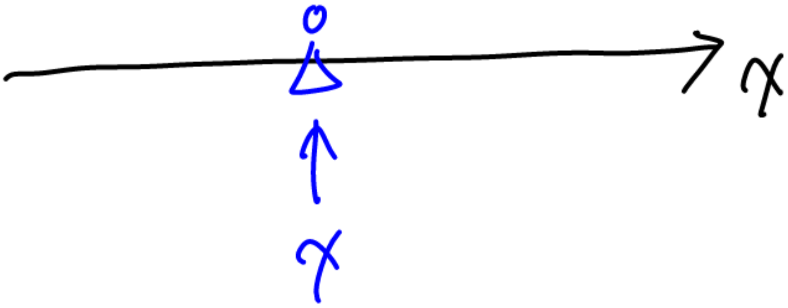
...   "nothing"

* Global vs Local

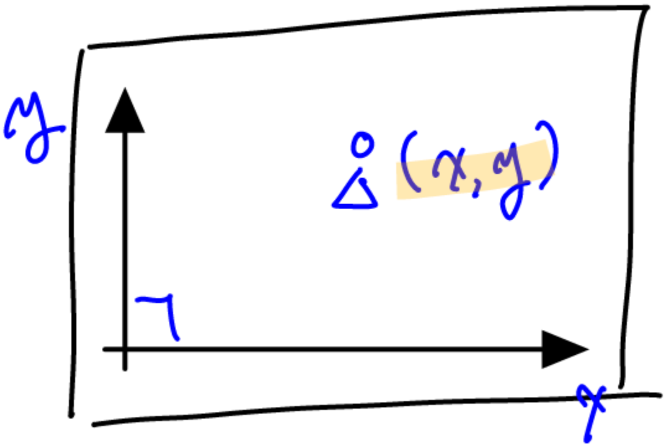
static, extern

* Dimension.

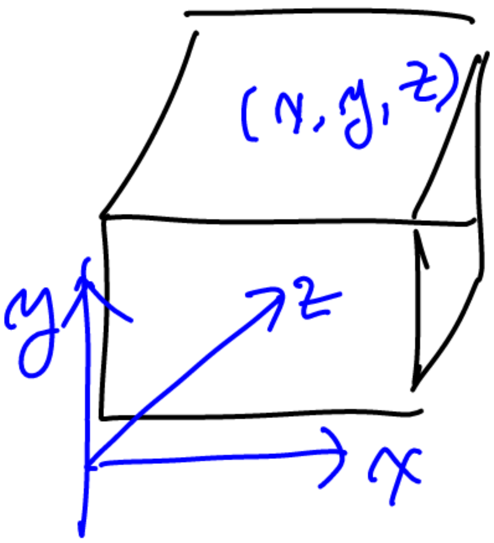
1 dimension



2 dimension



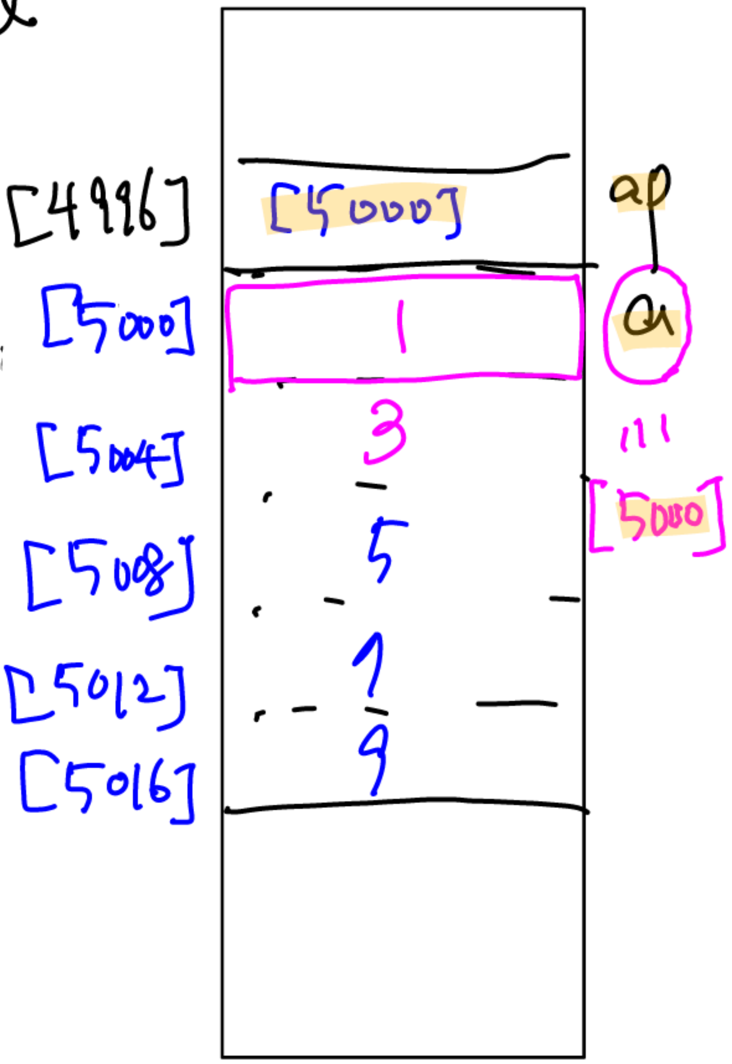
3 dimension.



形 (x, y, height, t ...)

* array. 1 dimensional

```
int a;
int aφ, a1, a2, a3
int a[5];
```



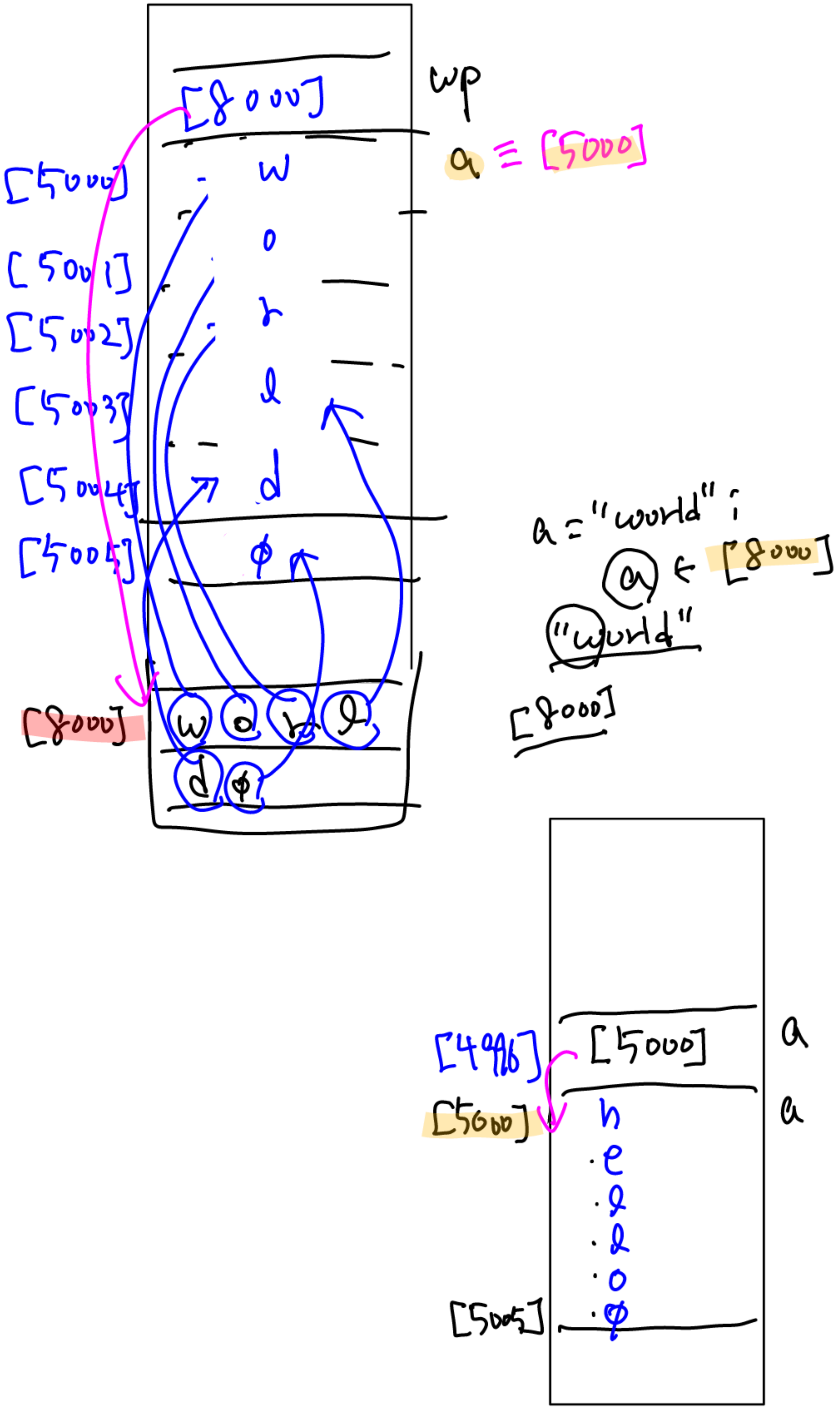
$*(a+n) = a[n]$

$n[a]$

[] operator.

* char array.

(page5)



[8000]
wp
a ≡ [5000]
[5000]
[5001]
[5002]
[5003]
[5004]
[5005]
w
o
r
l
d
φ
[8000]
w o r l
d φ
a = "world";
a ← [8000]
"world"
[8000]
[4996]
[5000]
a
[5000]
a
h
e
l
l
o
φ
[5005]

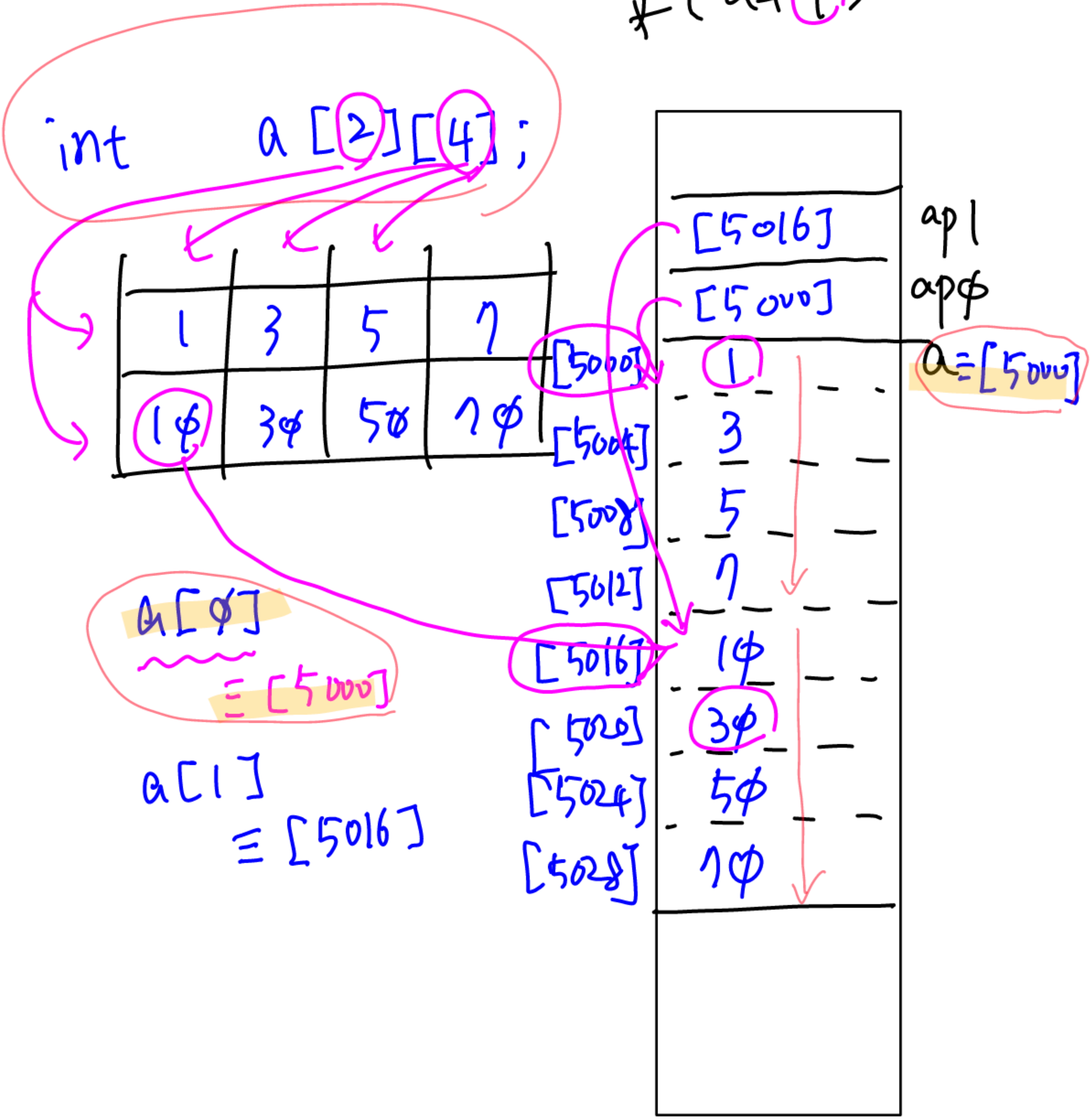
* 2 dimensional array

int a[5]; → 0 ~ 4

int: type
a: name
5: index

$*(a+0)$

$*(a+1)$

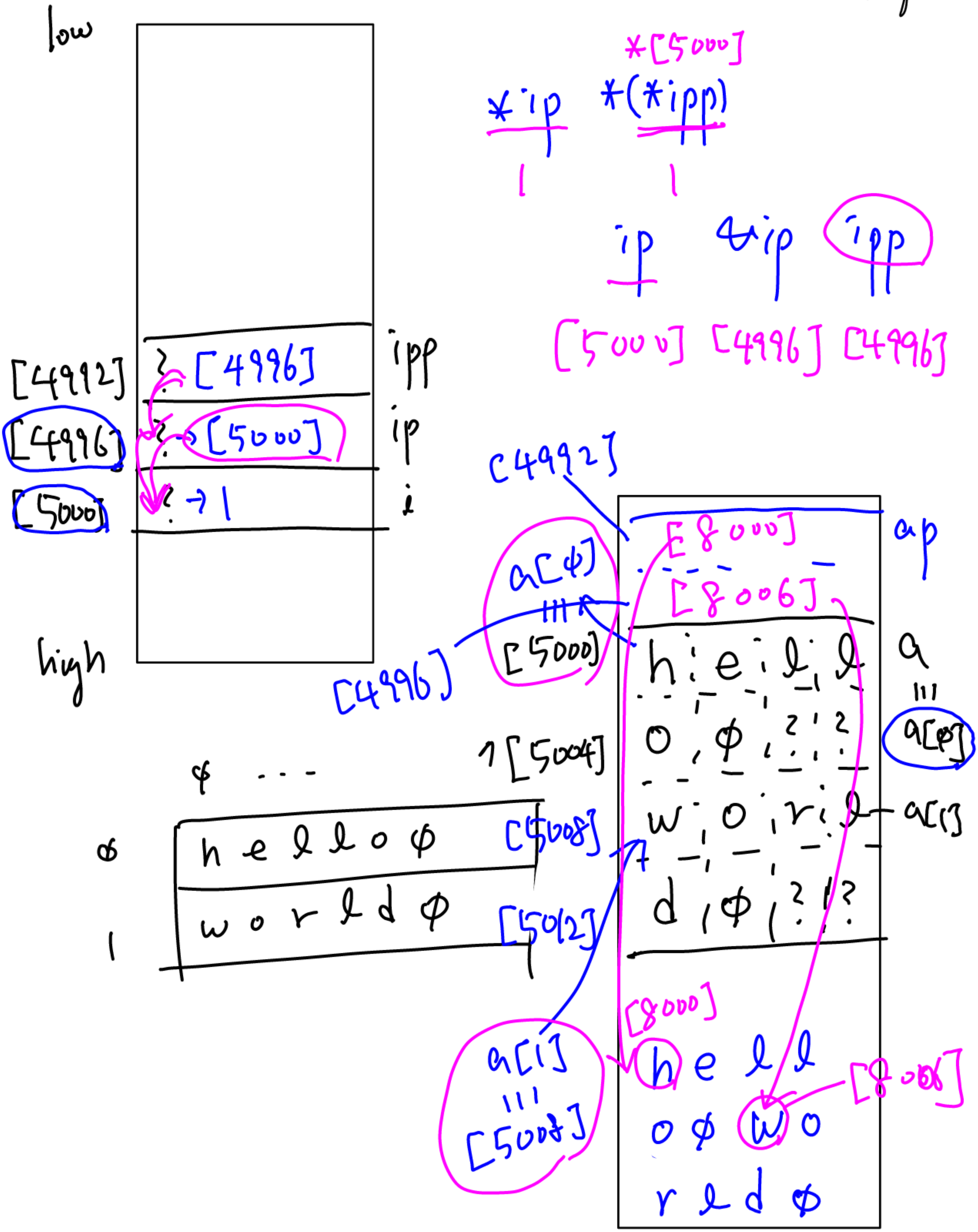


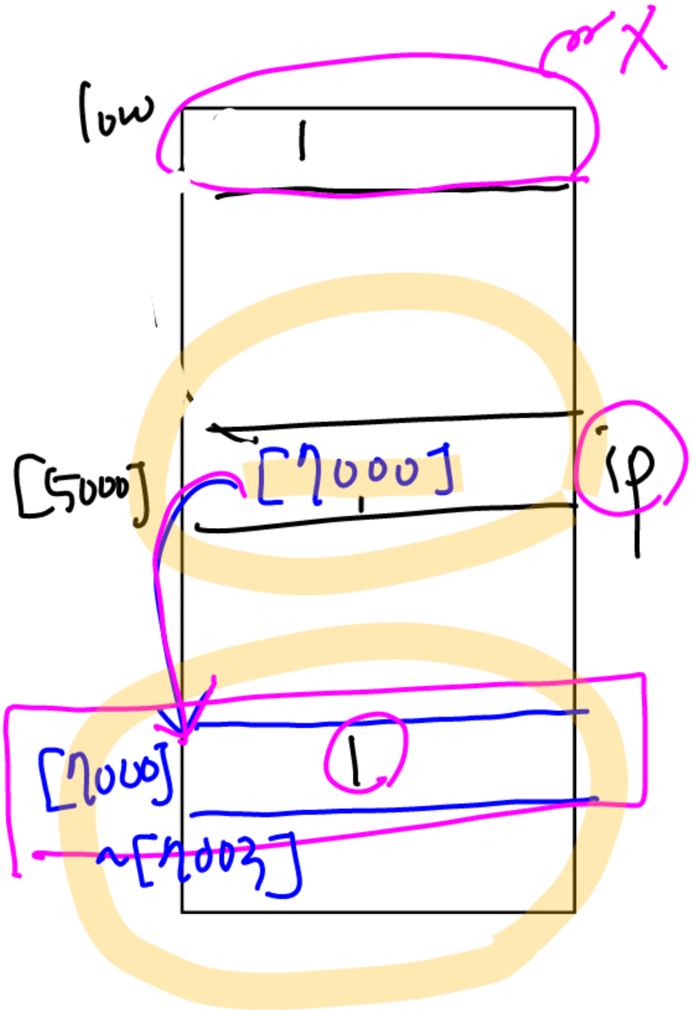
int a[2][4];

1	3	5	7
10	30	50	70

a[0] ≡ [5000]

a[1] ≡ [5016]

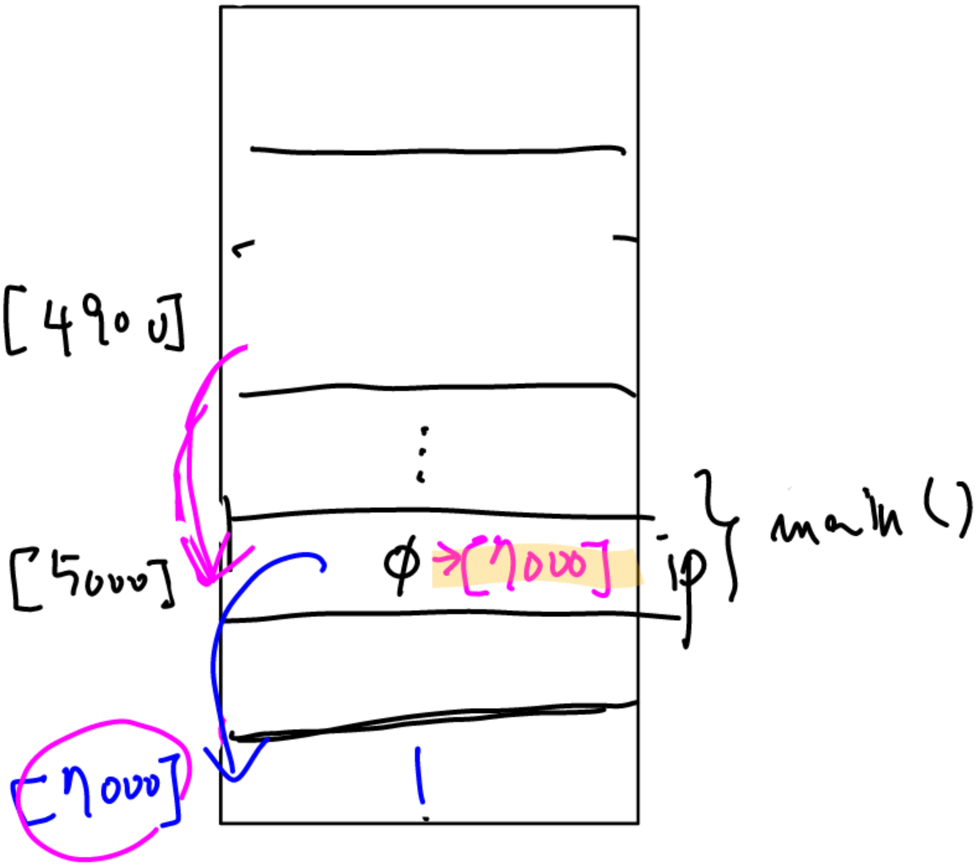

low
high
[4992]
[4996]
[5000]
? [4996]
? → [5000]
? → 1
ipp
ip
i
*[5000]
*ip
*(*ipp)
1
1
ip
&ip
ipp
[5000] [4996] [4996]
[4992]
ap
[8000]
[8006]
a[0]
[5000]
[4996]
a
a[0]
a[1]
h e l l
o 0 ? ?
w o r l
d 0 ? ?
[5004]
[5008]
[5012]
0 ...
0
1
h e l l o 0
w o r l d 0
[8000]
h e l l
o 0 w o
r l d 0
[8006]
a[1]
[5008]

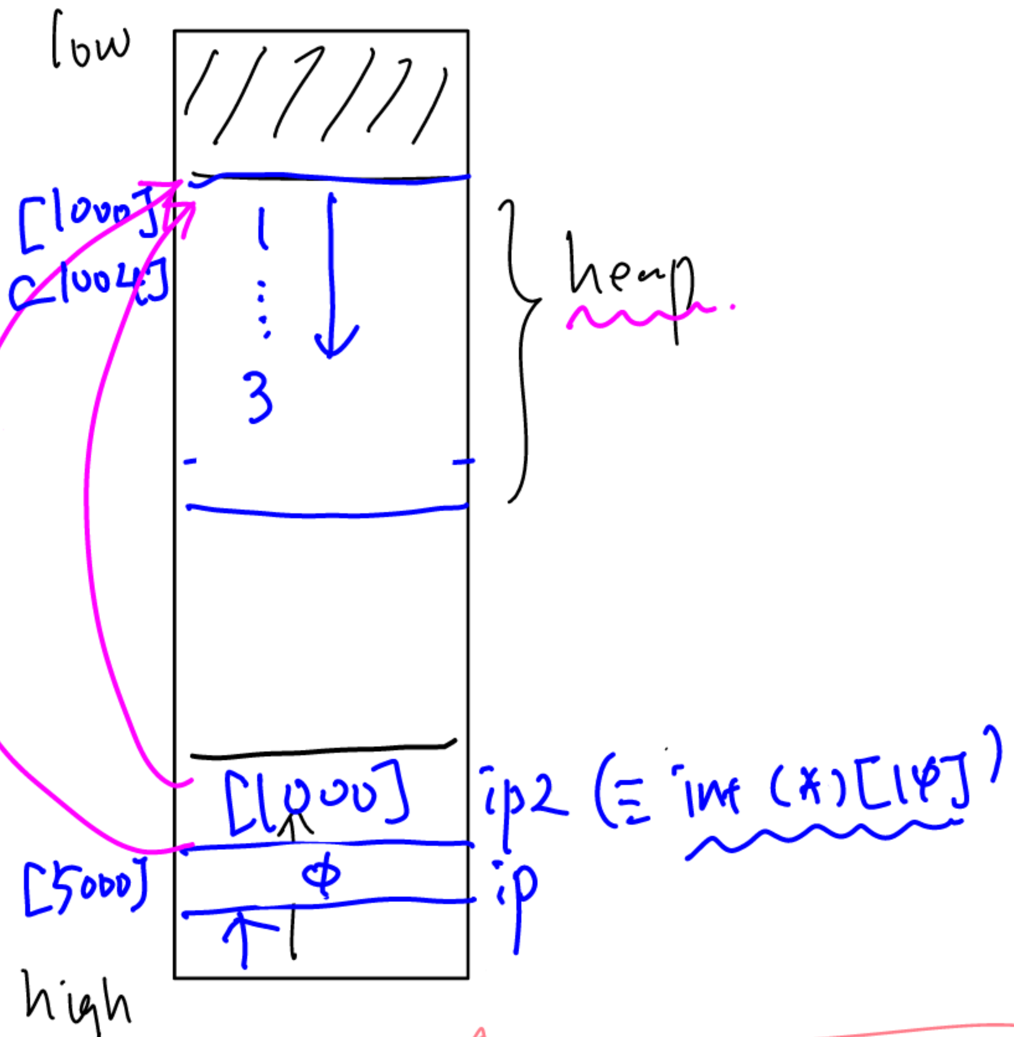


* Dynamic Memory Allocation

malloc ( )

free ( )





* type definition

typedef

enum

struct

union

class (C++ only)

unsigned char b;

byte

(page10)

```
typedef unsigned char byte;
typedef void (*Fun)( int);
```

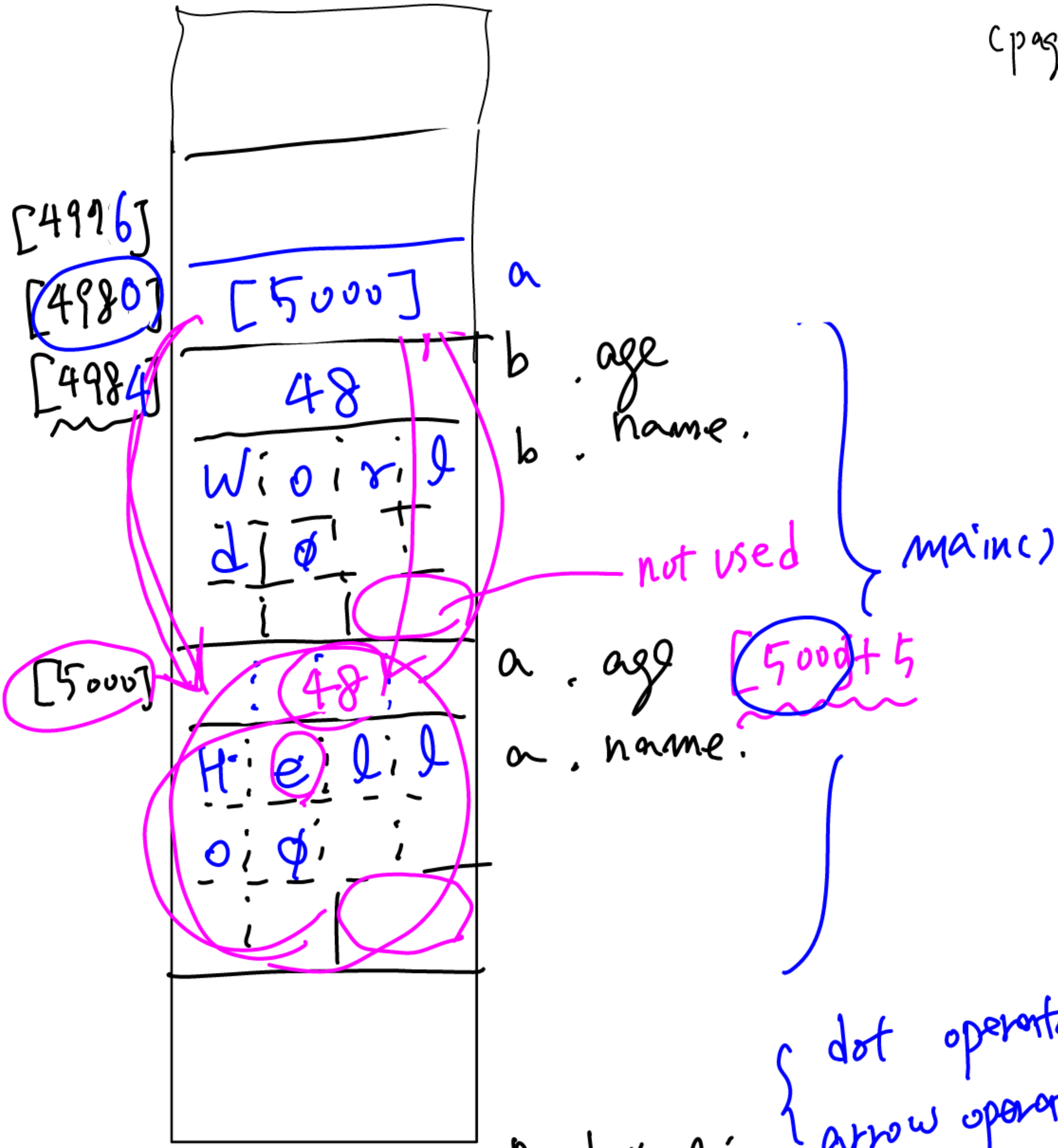
enum (enumeration)

```
enum tagName
{
    constant1[= value],
  [constant2[= value],][...]

};
```

* Structure.

```
struct
int a;
int a[10];
```



{ dot operator.
arrow operator.

People* a;

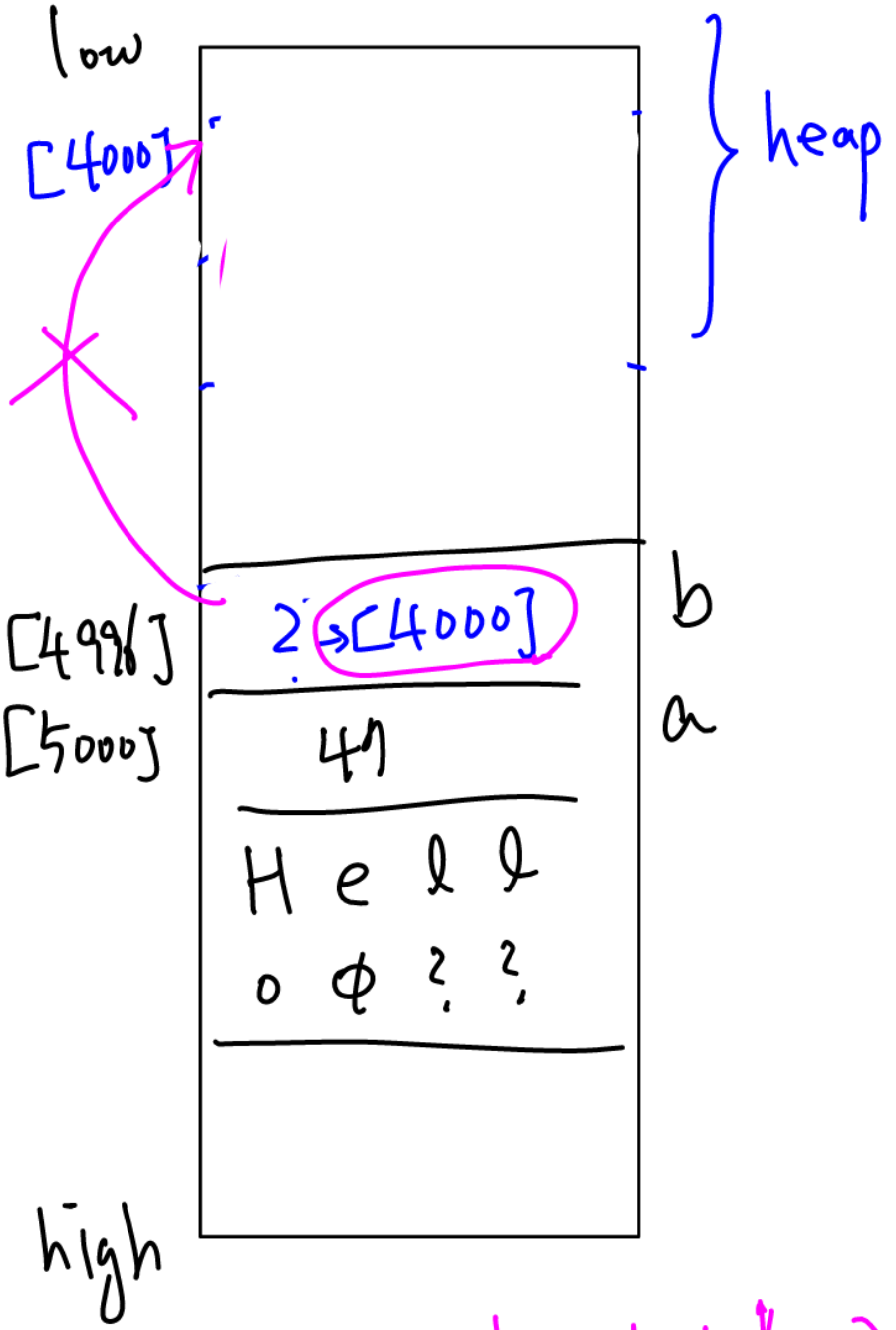
(*a).age =

|||

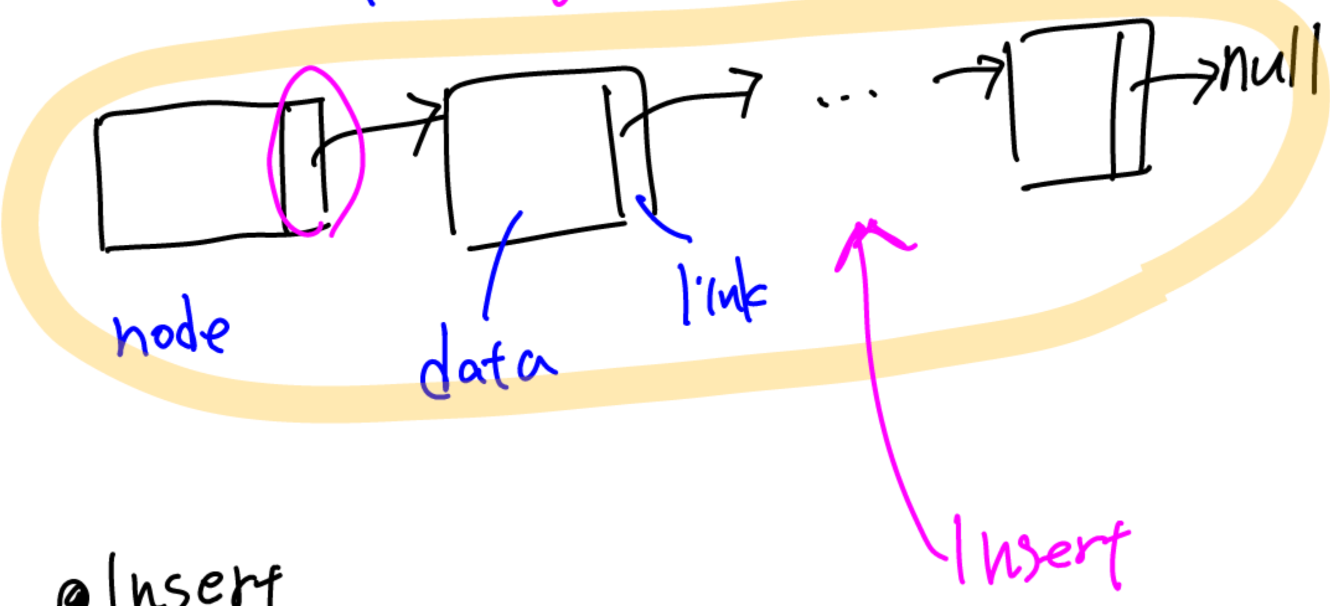
a->age;

a.age

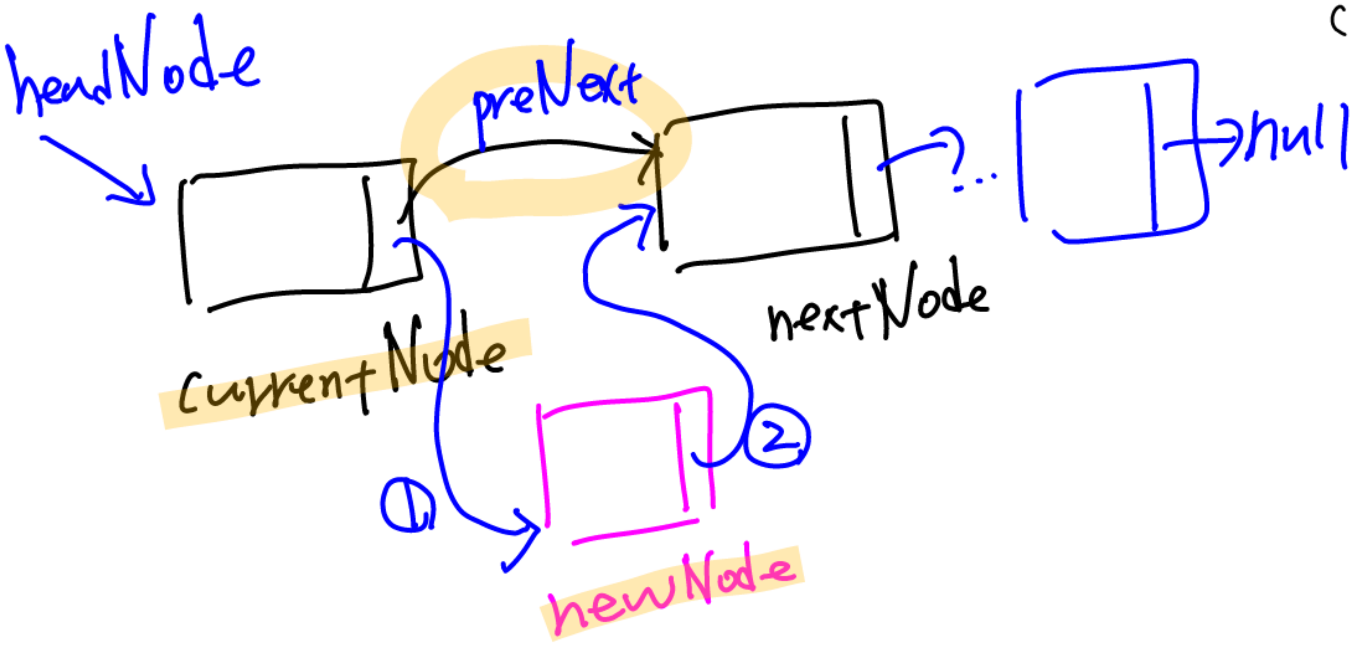
a.name



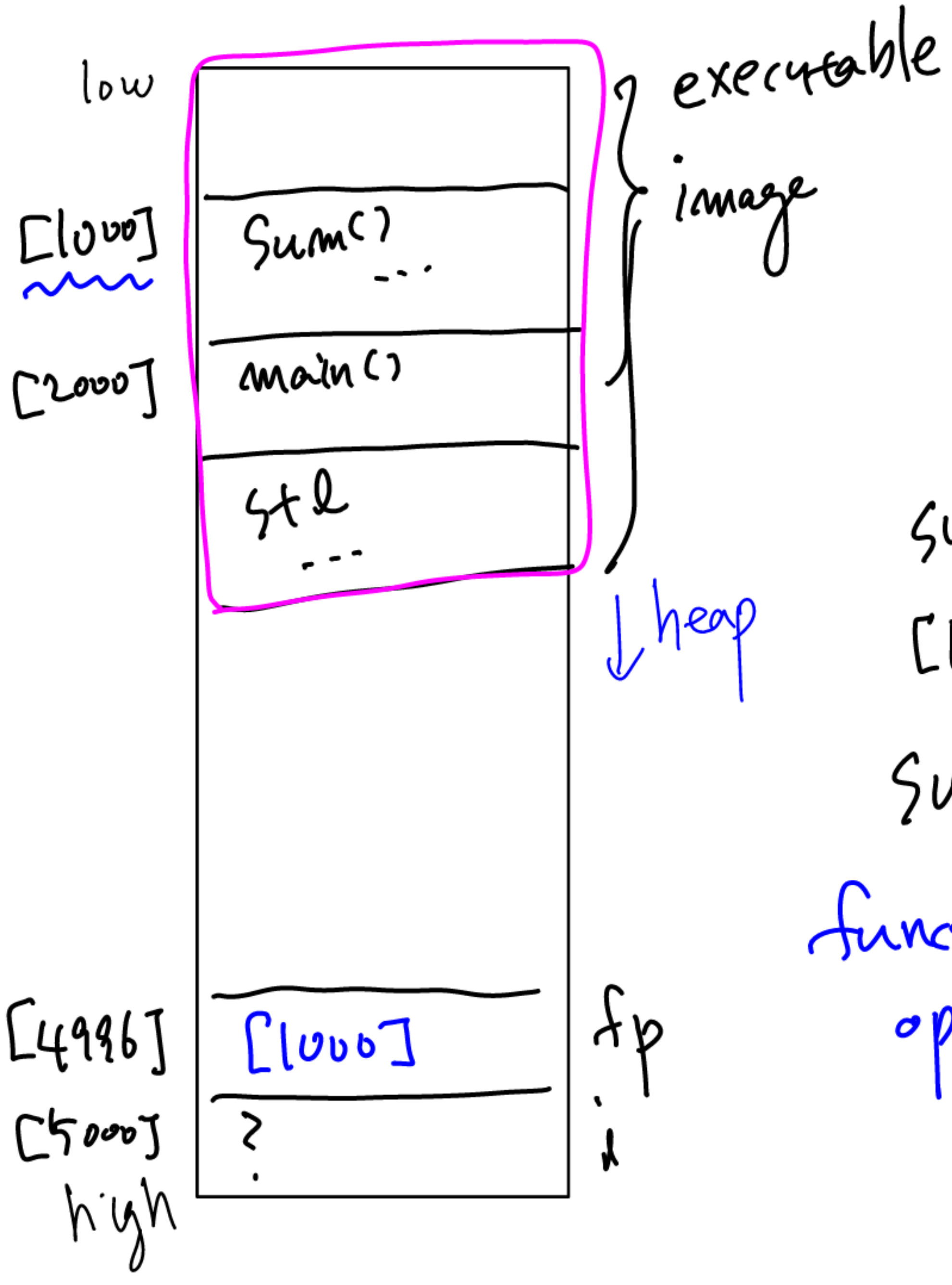
*Linked List (single linked list)



@Insert



* function pointer



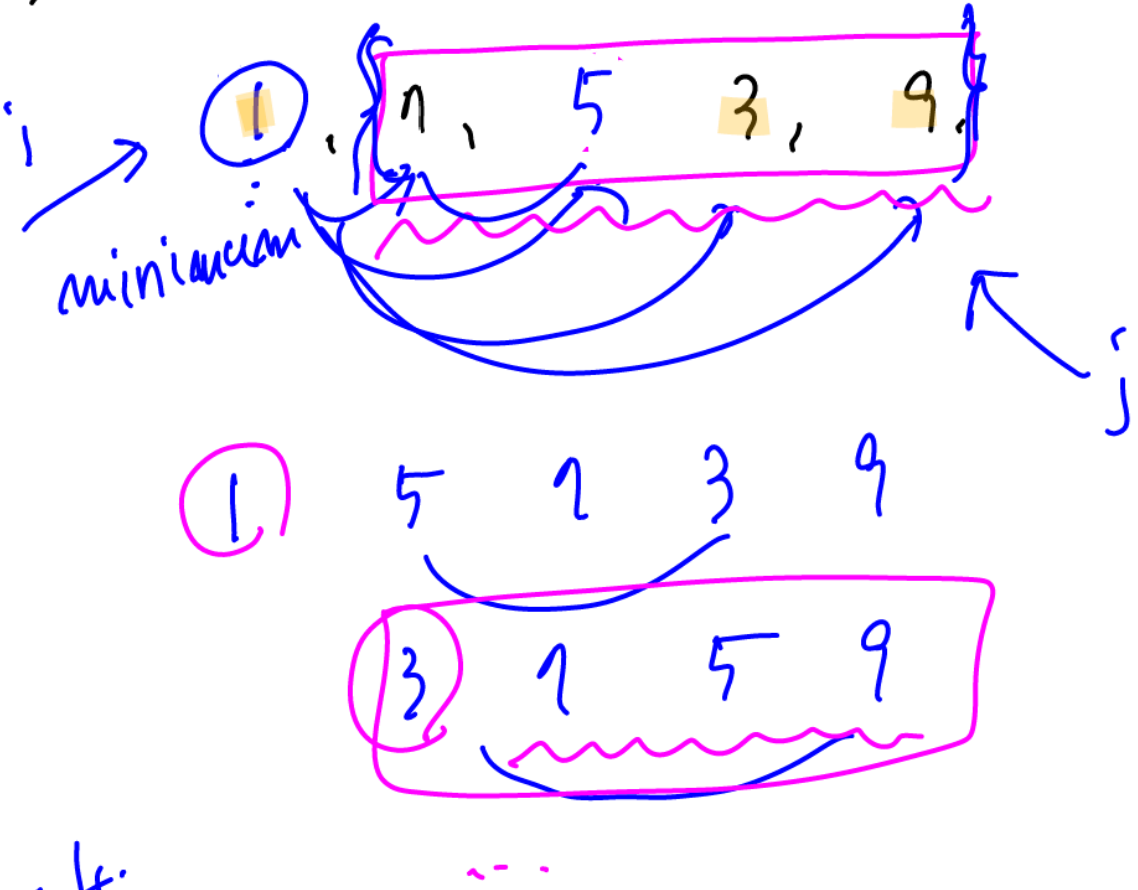
sum;

[1000];

sum()

function call operator.

* Selection Sort



result:

1 3 5 7 9

* preprocessing command

#include
#define
#if
#ifdef ...
...

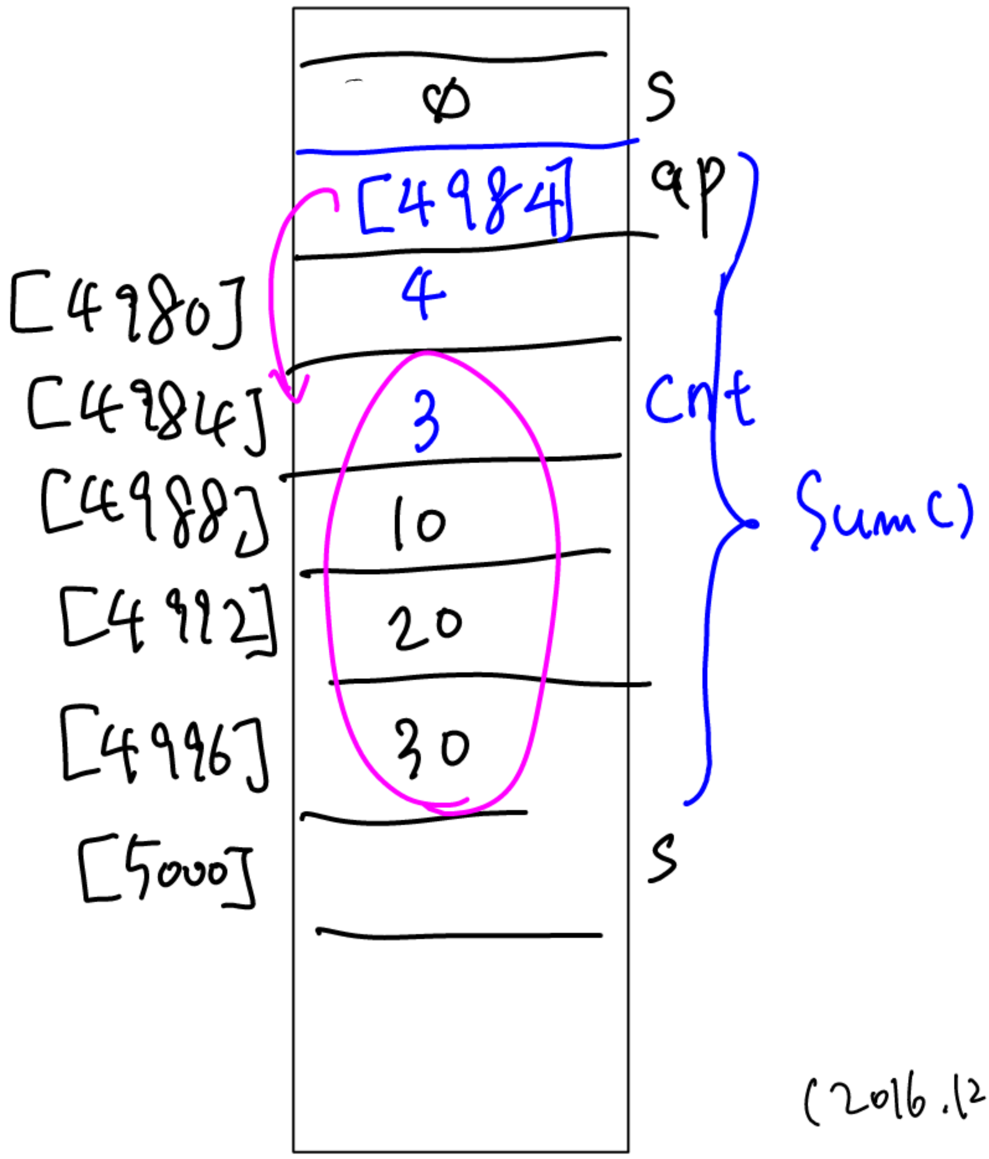
defined()
[illegible] !!

* predefined preprocessing command
(= compiler variable)
__LINE__
__FUNCTION__

##: token concatenation operator. (page 15)

* variable argument

...



(2016.12.22)