Stack & Heap

0x00	
0x01	
0x02	
0x03	
...	
0x1A	
0x1B	

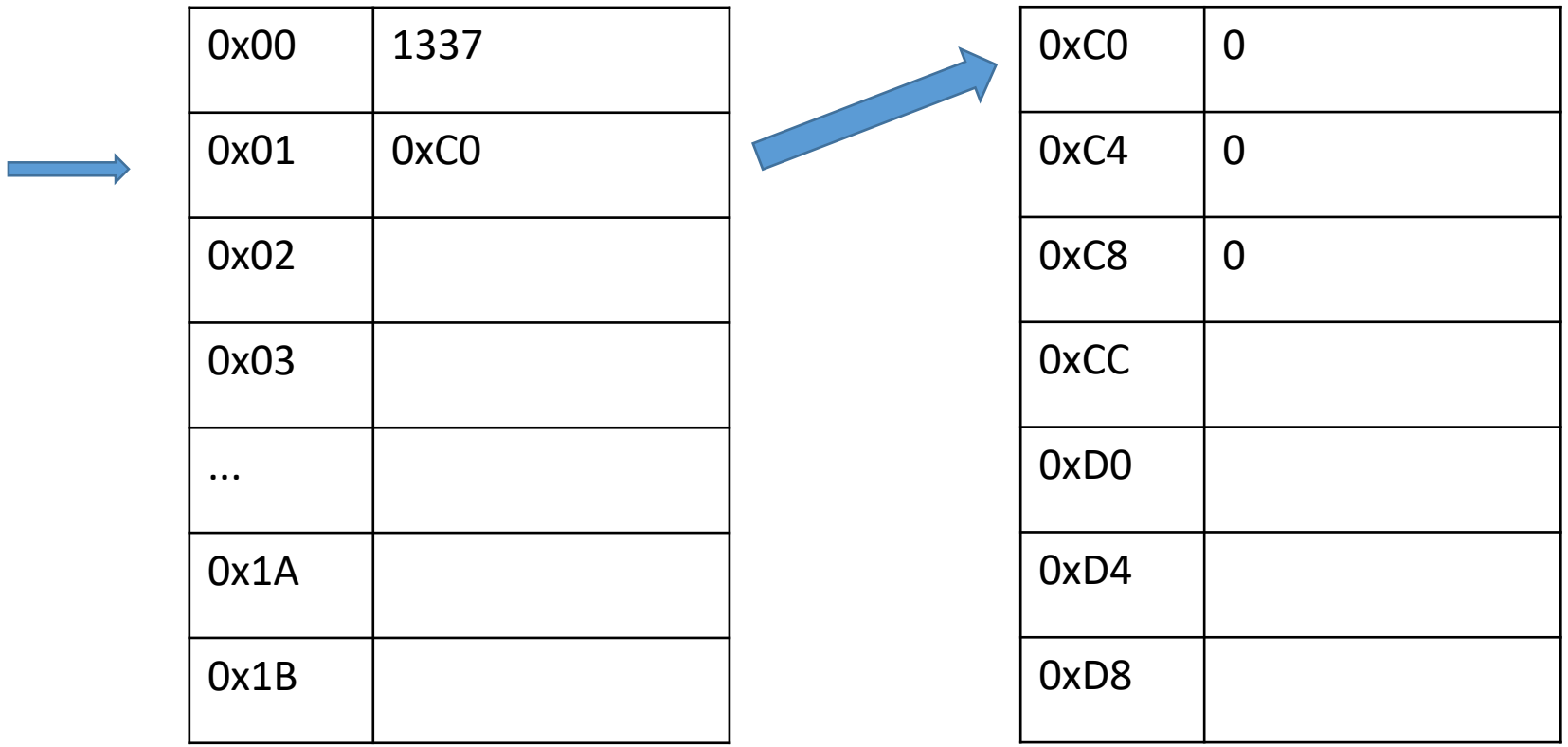
0xC0	
0xC4	
0xC8	
0xCC	
0xD0	
0xD4	
0xD8	

int i = 1337;

0x00	1337
0x01	
0x02	
0x03	
…	
0x1A	
0x1B	

0xC0	
0xC4	
0xC8	
0xCC	
0xD0	
0xD4	
0xD8	

int* heaparray = new int[3];



0x00	1337
0x01	0xC0
0x02	
0x03	
…	
0x1A	
0x1B	

0xC0	0
0xC4	0
0xC8	0
0xCC	
0xD0	
0xD4	
0xD8	

delete[](heaparray);

heaparray[1] = 3;

0x00	1337
0x01	0xC0
0x02	
0x03	
...	
0x1A	
0x1B	

0xC0 +
1*sizeof(int)

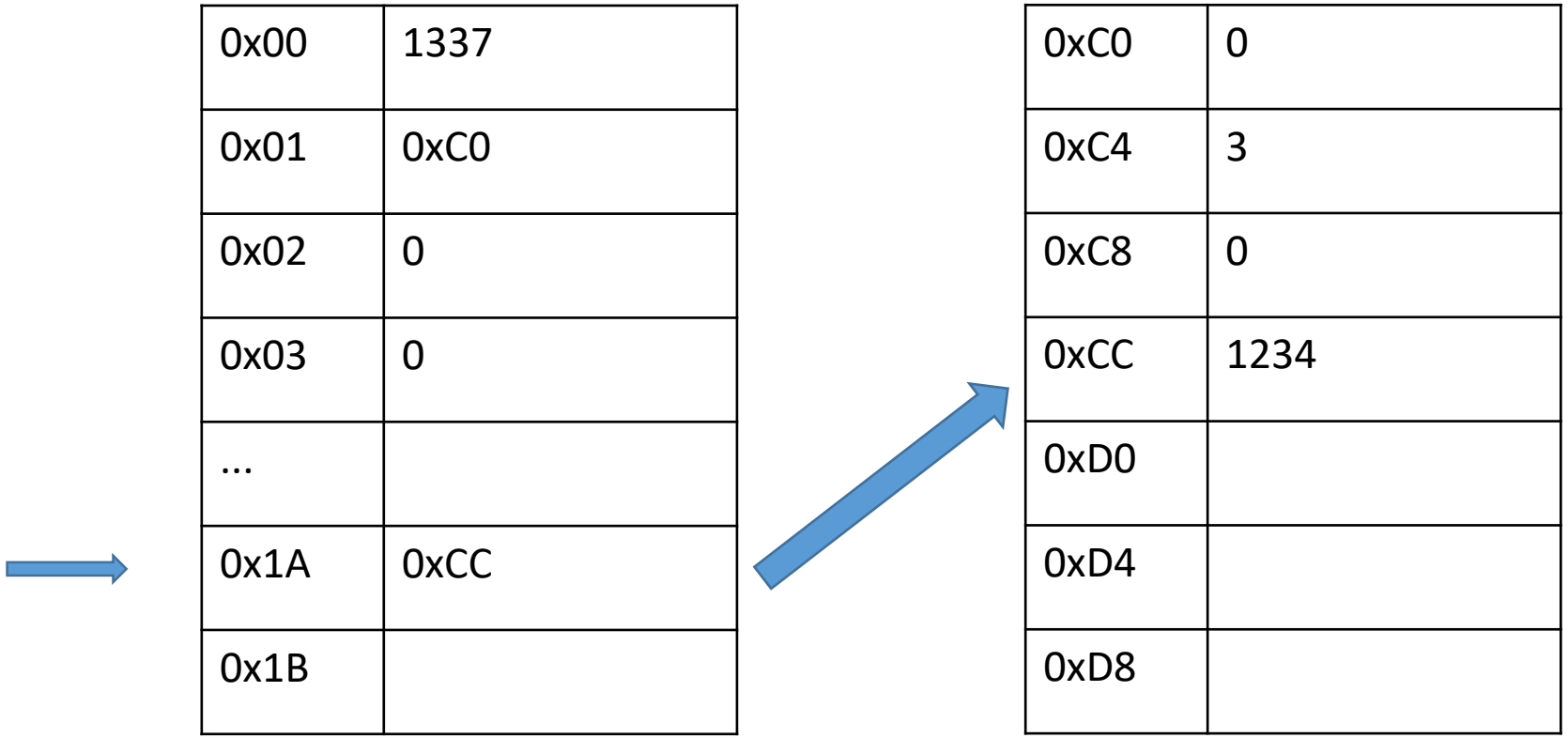
0xC0	0
0xC4	3
0xC8	0
0xCC	
0xD0	
0xD4	
0xD8	

int stackarray[2];

0x00	1337
0x01	0xC0
0x02	0
0x03	0
…	
0x1A	
0x1B	

0xC0	0
0xC4	3
0xC8	0
0xCC	
0xD0	
0xD4	
0xD8	

int* p = new int(1234);



0x00	1337
0x01	0xC0
0x02	0
0x03	0
…	
0x1A	0xCC
0x1B	

0xC0	0
0xC4	3
0xC8	0
0xCC	1234
0xD0	
0xD4	
0xD8	

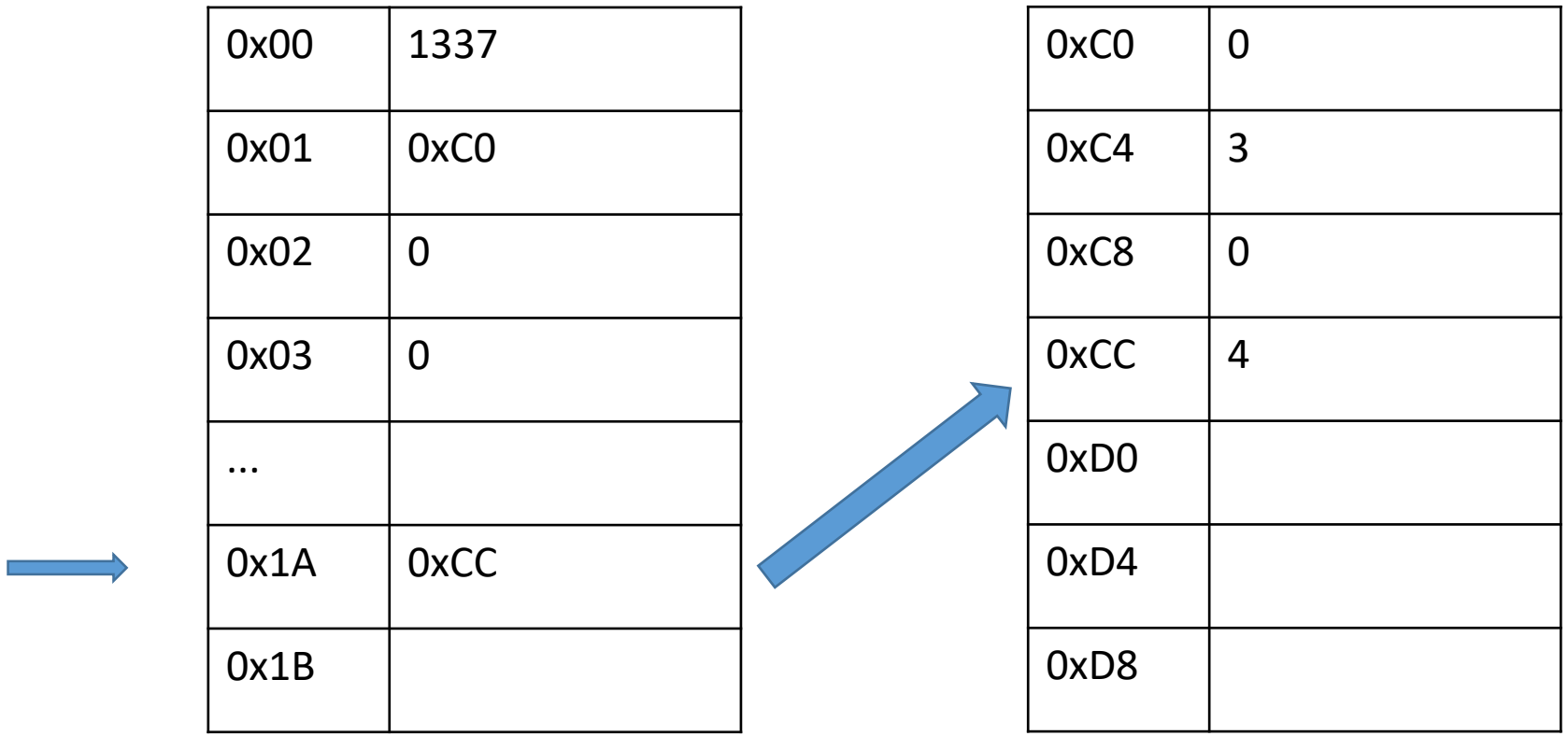
delete(p);

p = 4;

0x00	1337
0x01	0xC0
0x02	0
0x03	0
...	
0x1A	4
0x1B	

0xC0	0
0xC4	3
0xC8	0
0xCC	1234
0xD0	
0xD4	
0xD8	

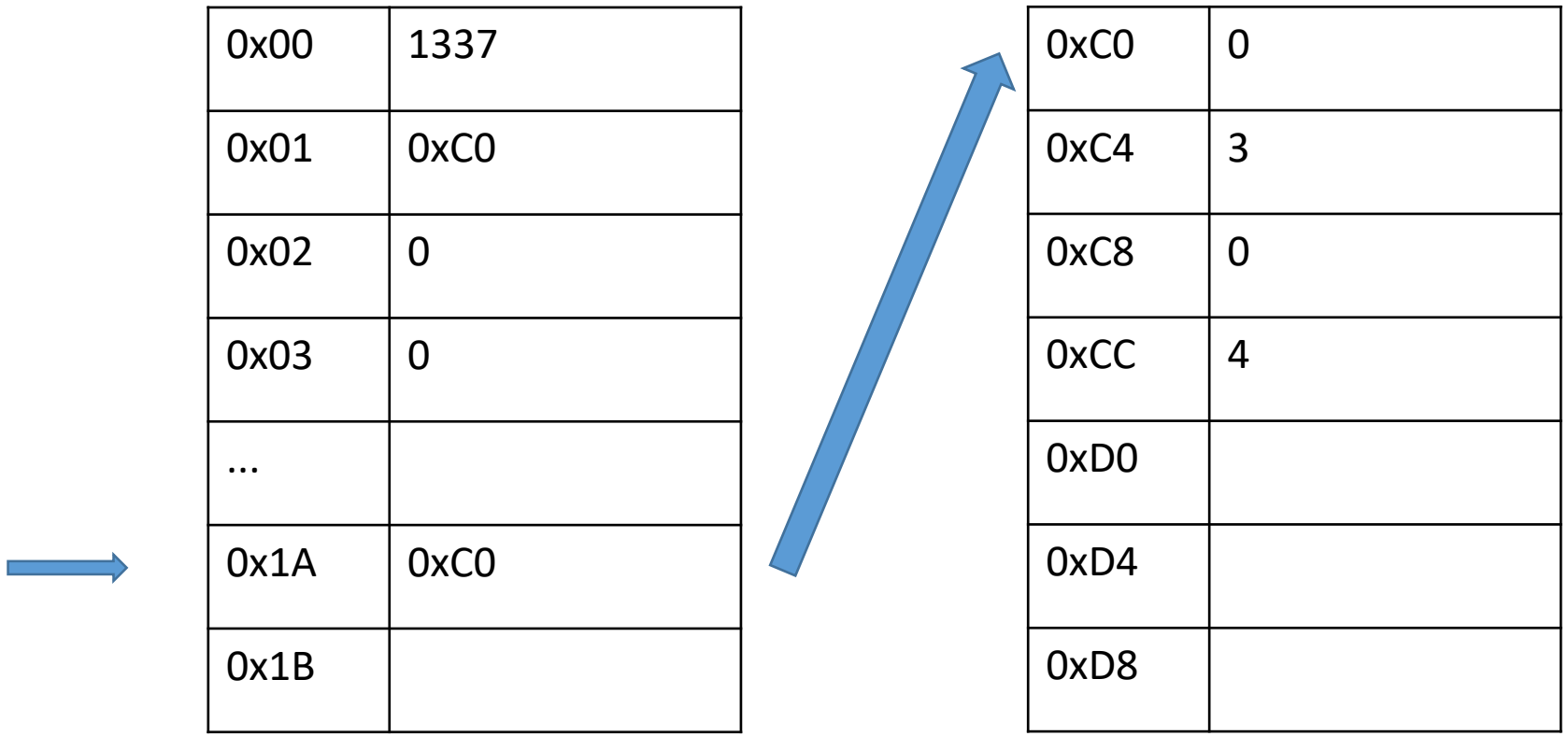
*p = 4;



0x00	1337
0x01	0xC0
0x02	0
0x03	0
…	
0x1A	0xCC
0x1B	

0xC0	0
0xC4	3
0xC8	0
0xCC	4
0xD0	
0xD4	
0xD8	

p = heaparray;

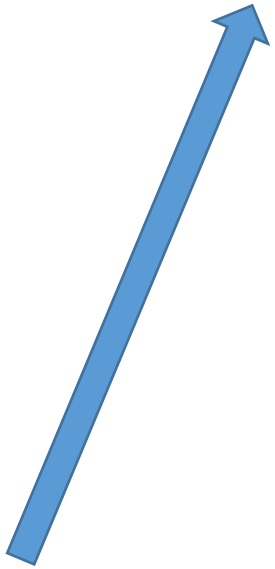


0x00	1337
0x01	0xC0
0x02	0
0x03	0
…	
0x1A	0xC0
0x1B	

0xC0	0
0xC4	3
0xC8	0
0xCC	4
0xD0	
0xD4	
0xD8	

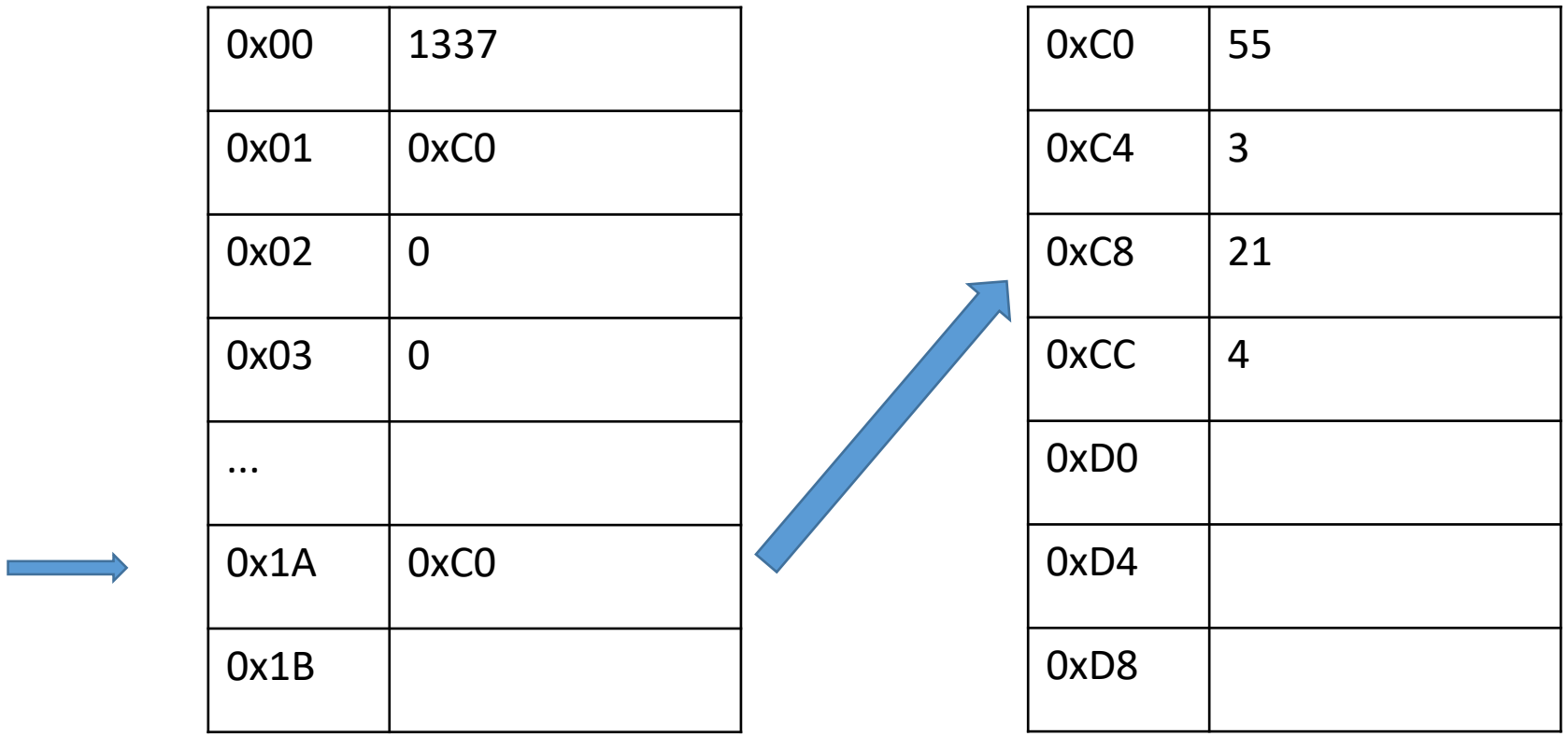
*p = 55;

0x00	1337
0x01	0xC0
0x02	0
0x03	0
…	
0x1A	0xC0
0x1B	



0xC0	55
0xC4	3
0xC8	0
0xCC	4
0xD0	
0xD4	
0xD8	

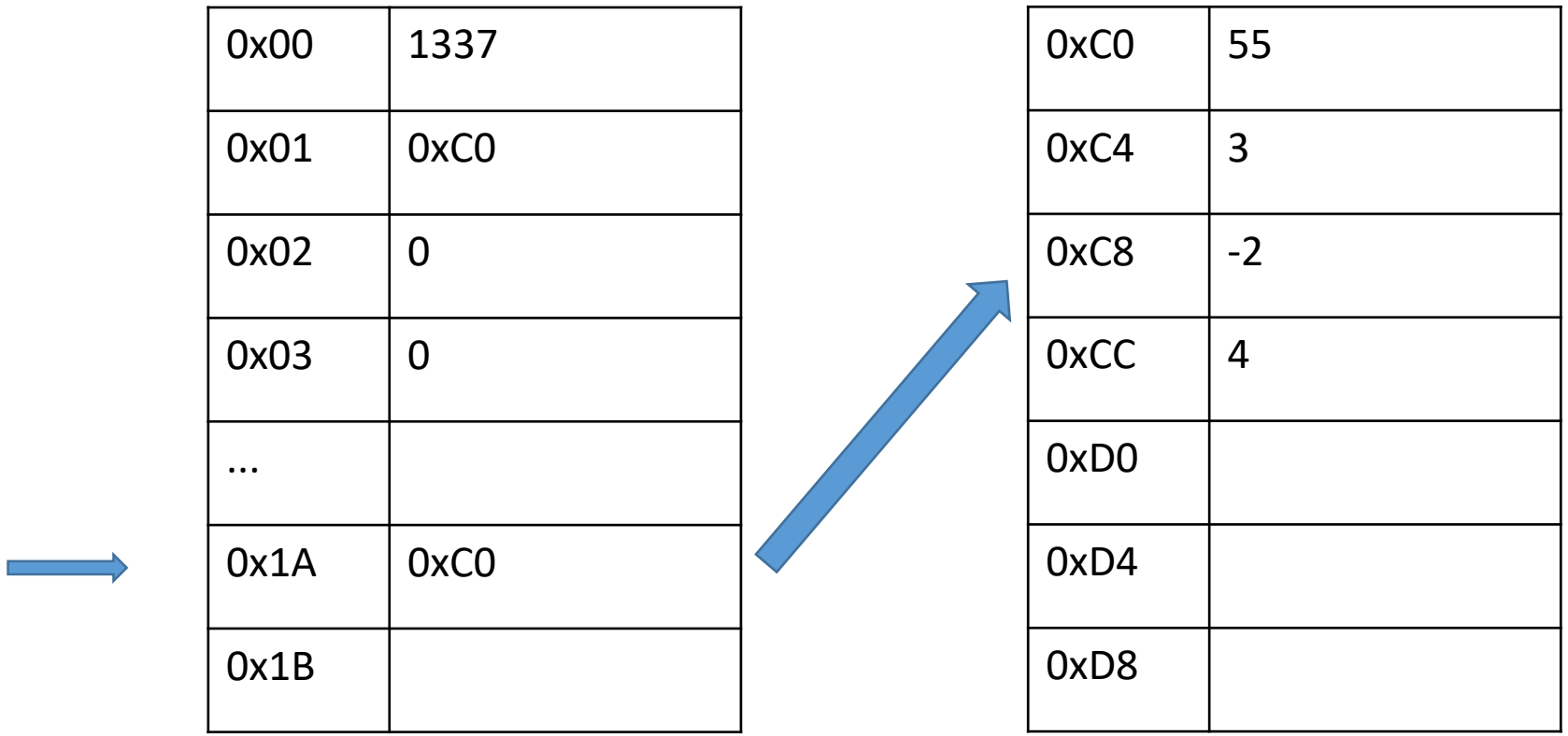
*(p+2) = 21;



0x00	1337
0x01	0xC0
0x02	0
0x03	0
...	
0x1A	0xC0
0x1B	

0xC0	55
0xC4	3
0xC8	21
0xCC	4
0xD0	
0xD4	
0xD8	

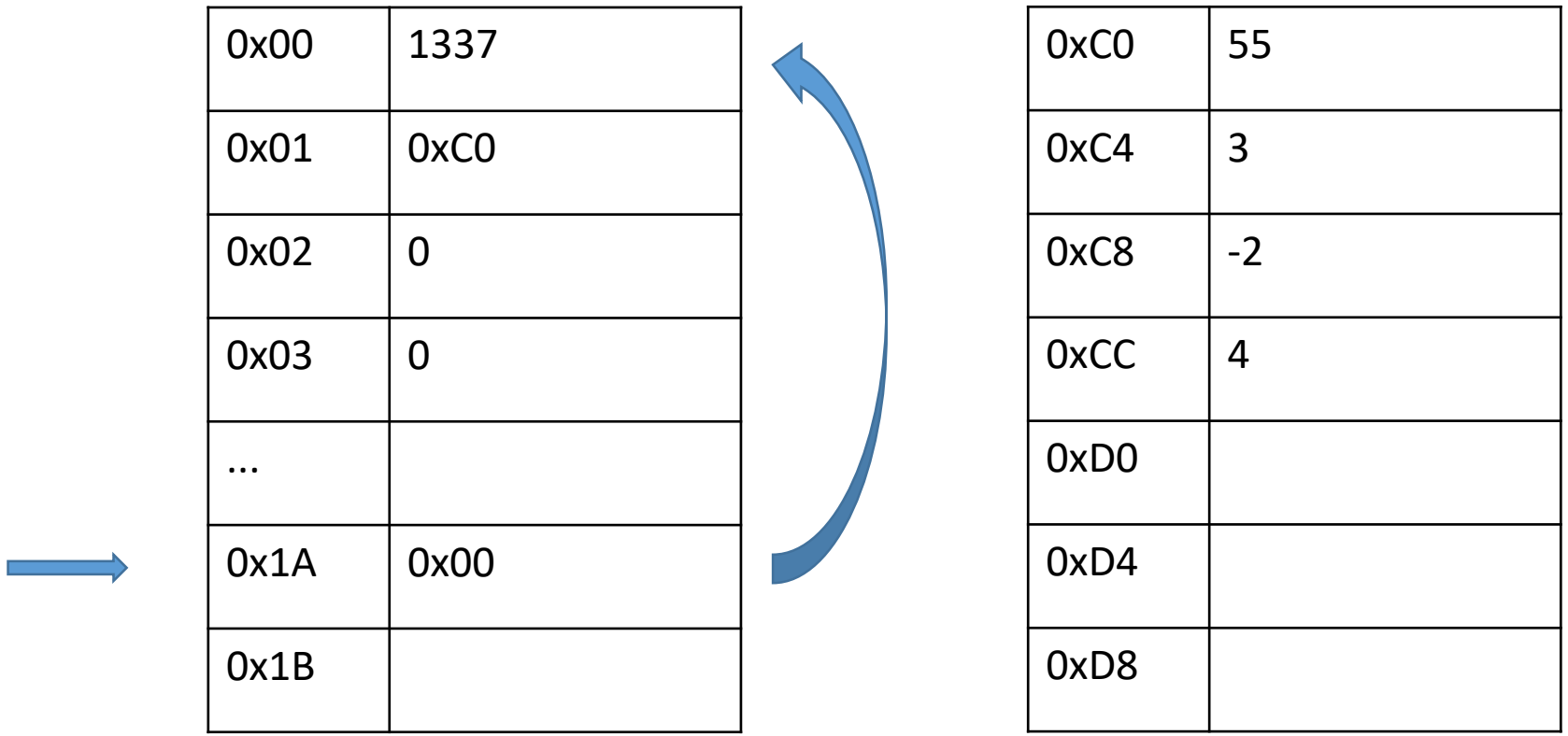
p[2]= -2;



0x00	1337
0x01	0xC0
0x02	0
0x03	0
...	
0x1A	0xC0
0x1B	

0xC0	55
0xC4	3
0xC8	-2
0xCC	4
0xD0	
0xD4	
0xD8	

p = &i;



0x00	1337
0x01	0xC0
0x02	0
0x03	0
...	
0x1A	0x00
0x1B	

0xC0	55
0xC4	3
0xC8	-2
0xCC	4
0xD0	
0xD4	
0xD8	

call-by-value/call-by-copy

```
void swap(int a, int b){
        int temp = a;
        a = b;
        b = temp;
}
Int main(){
        int a = 2;
        int b = 1;
        swap(a, b);
        std::cout << a << b;
}
```

Output:
21

call-by-reference

```
void swap(int* a, int* b){
        int temp = *a;
        *a = *b;
        *b = temp;
}
Int main(){
        int a = 2;
        int b = 1;
        swap(&a, &b);
        std::cout << a << b;
}
```

call-by-reference

```
void swap(int* a, int* b){
        int temp = *a;
        *a = *b;
        *b = temp;
}
Int main(){
        int a = 2;
        int b = 1;
        swap(&a, &b);
        std::cout << a << b;
}
```

Output:
12

call-by-reference 2

```
void swap(int &a, int &b){
        int temp = a;
        a = b;
        b = temp;
}
Int main(){
        int a = 2;
        int b = 1;
        swap(a, b);
        std::cout << a << b;
}
```

Output:
12