Revision
new
delete

Reference
- Alias for an existing memory location

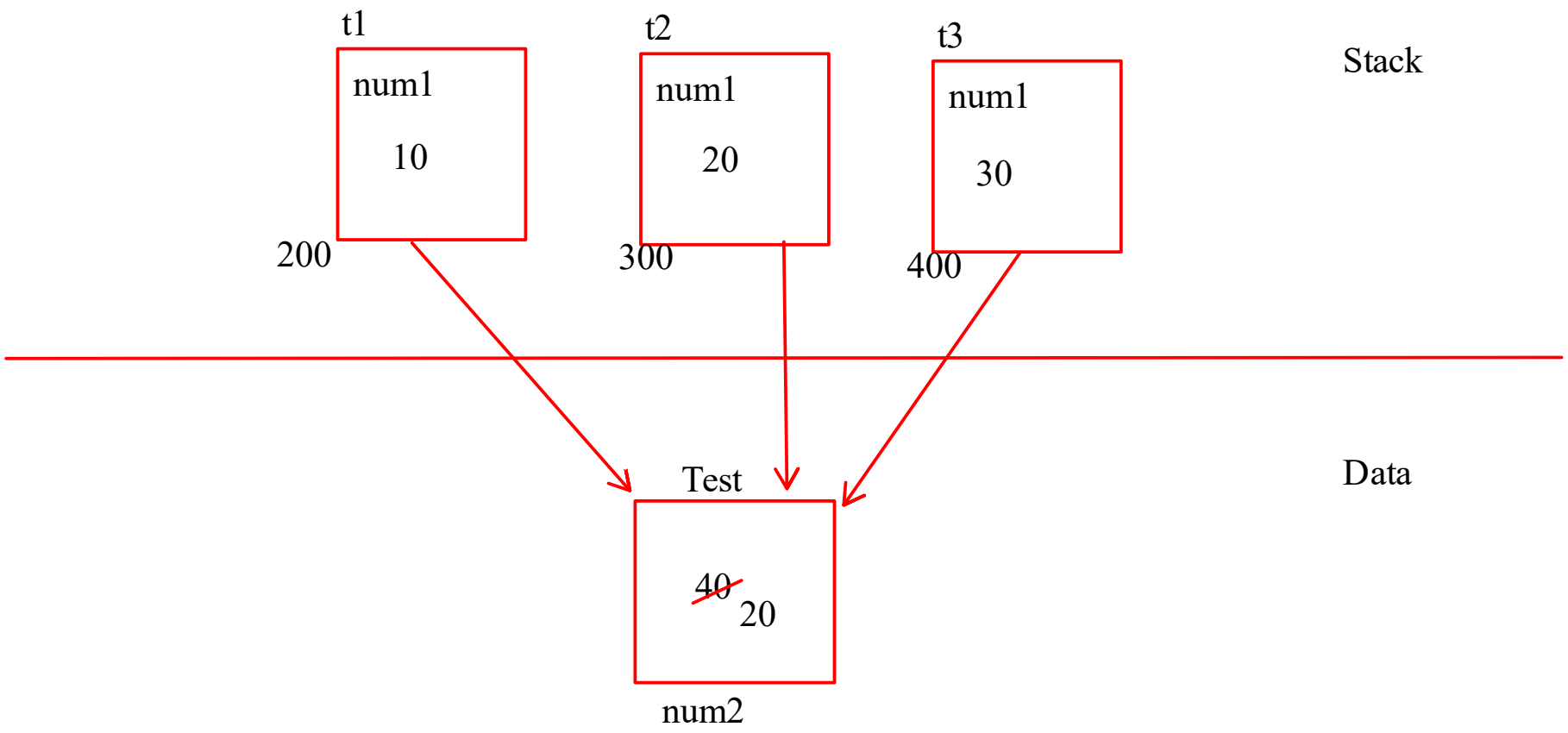
static
Array 1d,2d
enum

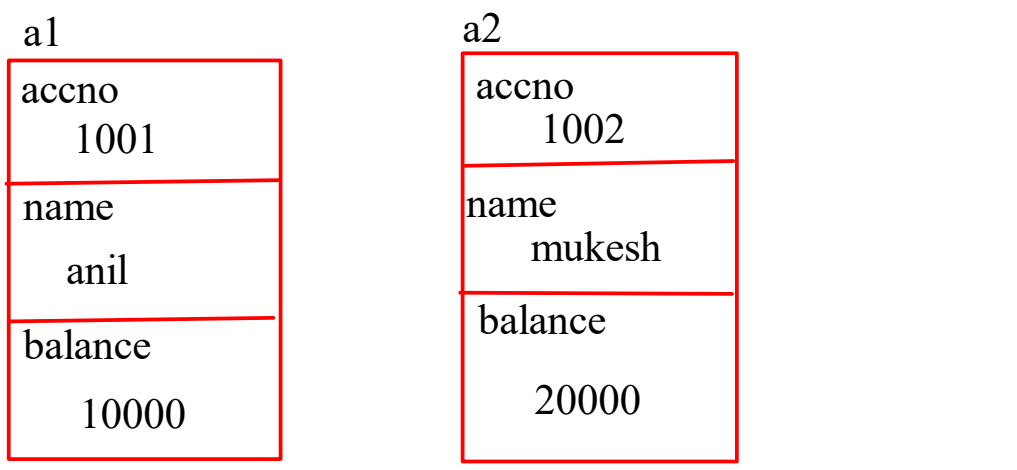
static ->

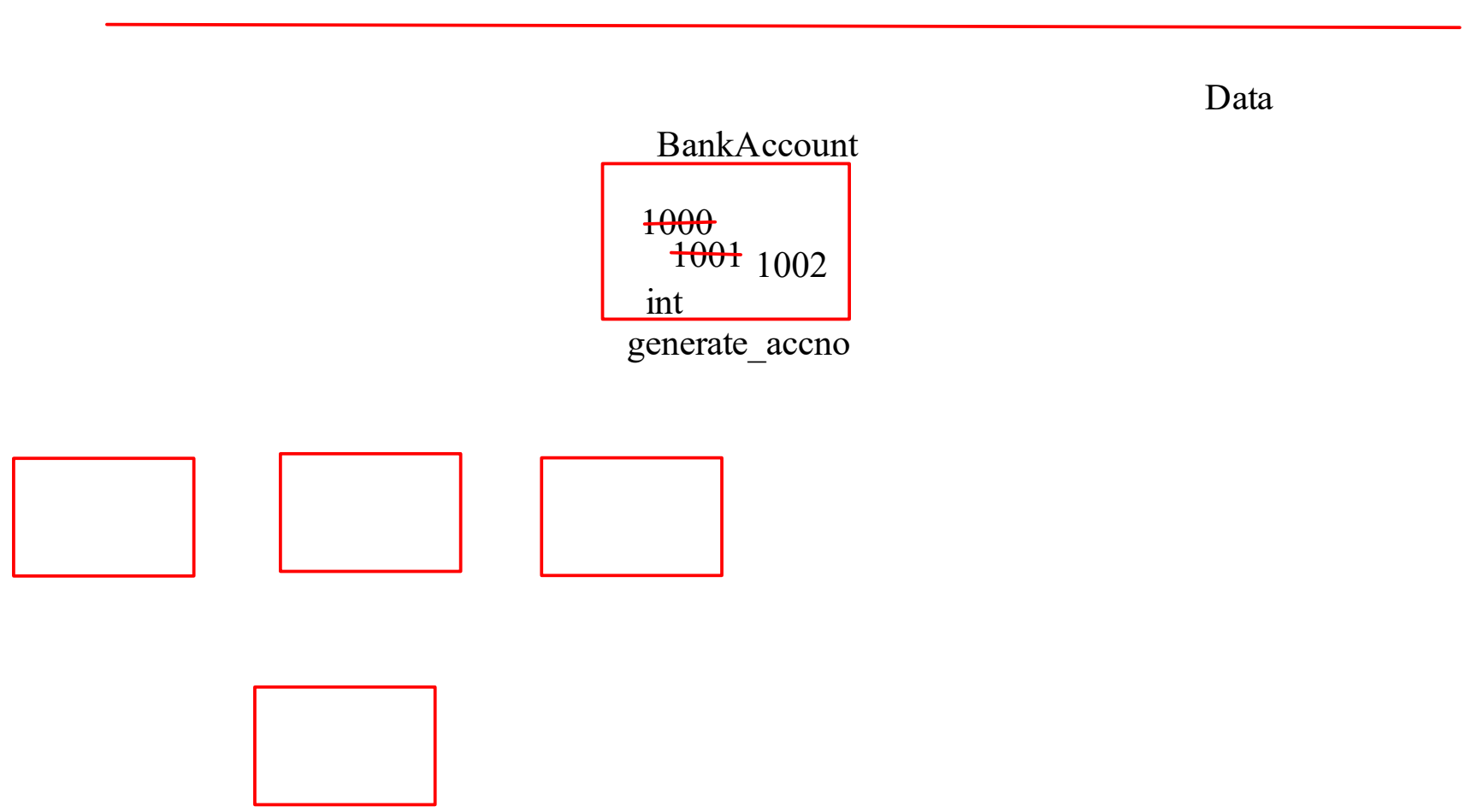
```
void f1(){
static int num1 = 10;
}

int main(){

}
```

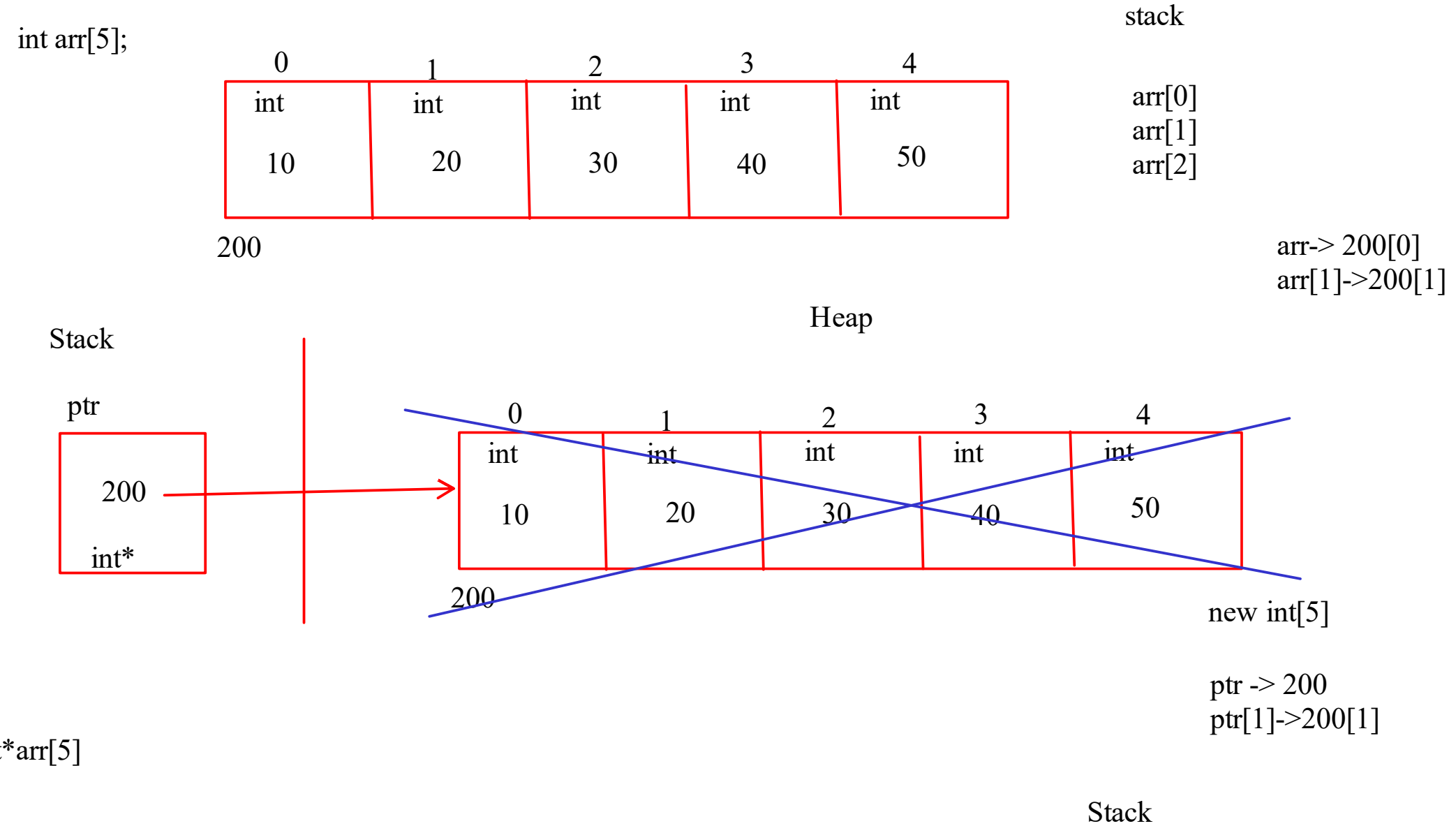






Static
- We can make datamemebrs and member functions as static
- Data members as satic
    - These are used to share between multiple objects
    - They get the memory only once at the time of program loading
    - Their initialization should be done outside the class on global scope using classname ::
    - We can make then as constant
- Member functions as static
    - These are designed to be called on class name using ::
    - These functions do not get this pointer
    - As this pointer is not given to static member function, we cannot access non static data memebers inside them
    we can access only static data members inside them.
    - we cannot make them as constant, as they do not have this pointer

Array
- It is a data structure/ collection used to store the data of similar types in contigious memory location
- Its size is fixed.



stack

ptr

200

int**

Heap

0 1 2 3 4

int* int* int* int* int*

500 300 400 350 600

step-2 -> delete the array

new int*[5]

200

200[0]=new int(10);

10 20 30 40 50

500 new int 300 new int 400 new int 350 new int 600 new int

200[0]->500

*ptr[0]

step-1 to delete these elements

delete ptr[0]

delete[] ptr;

Employee arr[5];

arr

0 1 2 3 4

Employee Employee Employee Employee Employee

empid 1 | name anil | salary 10000

empid 2 | name mukesh | salary 20000

empid | name | salary

empid | name | salary

empid | name | salary

200

200[0].acceptData();
200[0].displayData();

Employee* arr[5];

Stack

arr

0 1 2 3 4

Employee* Employee* Employee* Employee* Employee*

500 400

200

Heap

empid | name | salary

500

empid | name | salary

400

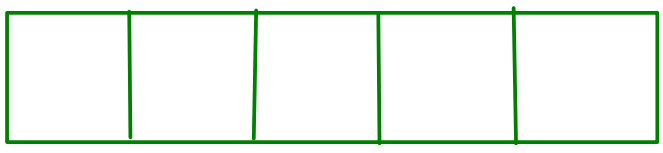
arr[0]= new Employee();

200[0]->

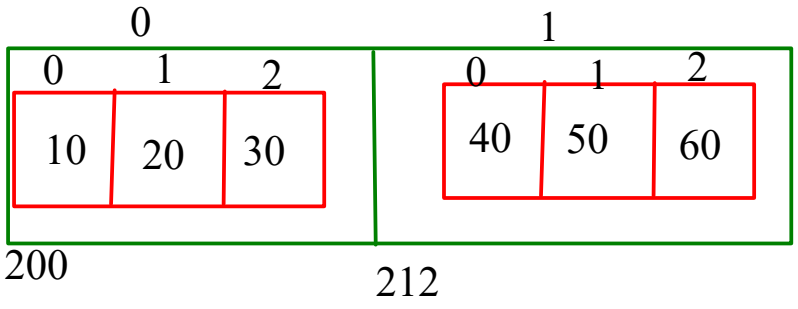
new Employee; new Employee(2,"Mukesh",20000);

MultiDimension Array (2D)

int arr[5]



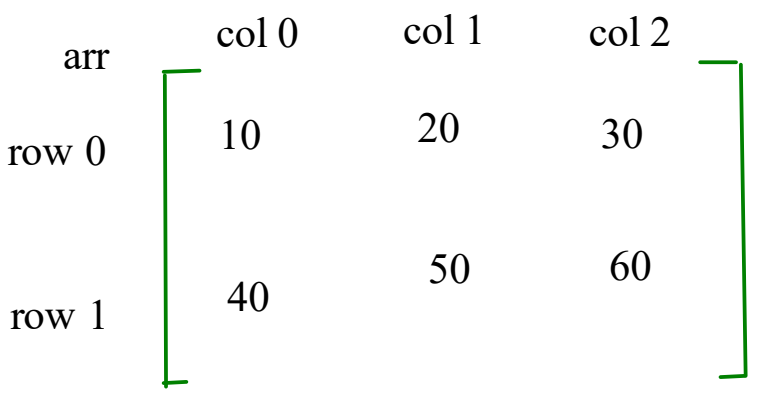
int arr[2][3]



arr[0]
arr[1]

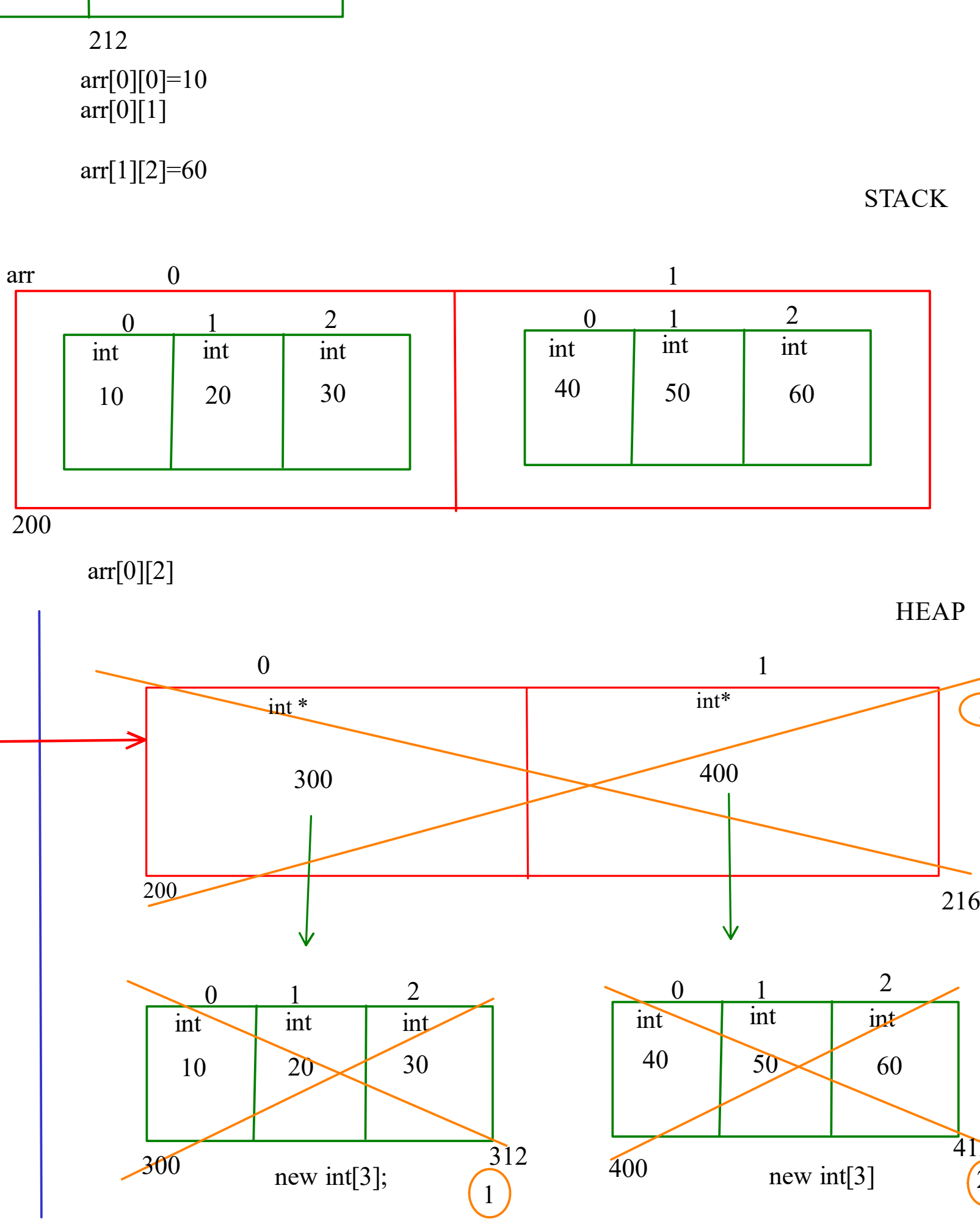
arr[0][0]=10
arr[0][1]

arr[1][2]=60



arr[row][col]
arr[1][0] -> 40

STACK



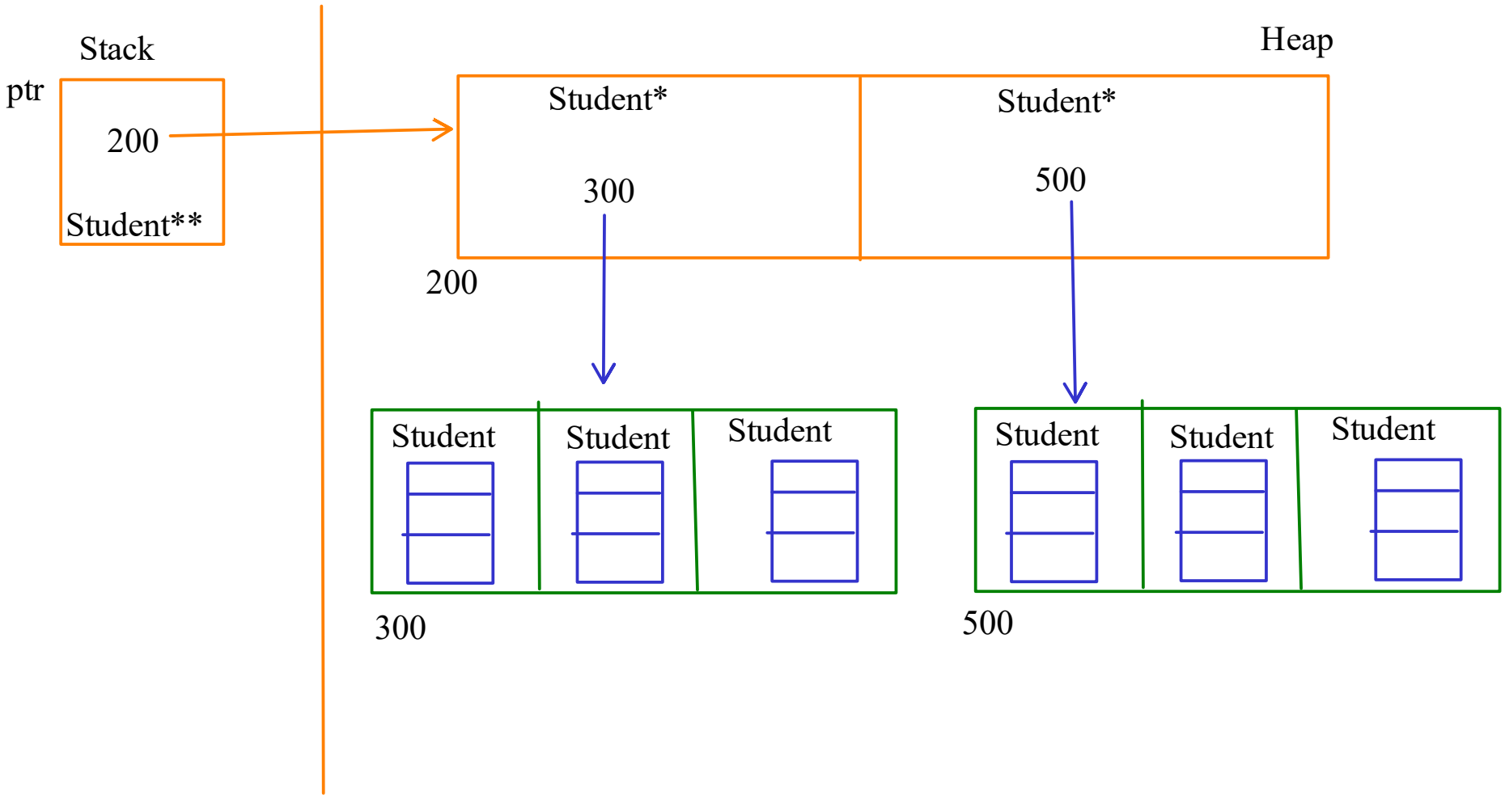
int *ptr = new int[3]

int* ptr = new int[2][3];

300 [0]
200[0] [0]
ptr[0][0]

delete [] ptr[0] -> delete [] 300

delete [] ptr[0];
delete [] ptr[1];
delete [] ptr;


Stack
ptr
200
Student**
Heap
Student*
300
Student*
500
200
Student
Student
Student
300
Student
Student
Student
500