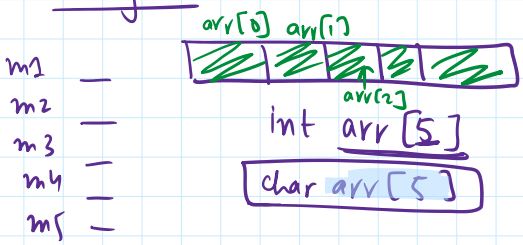
Class 5

Sunday, June 3, 2018 8:32 AM

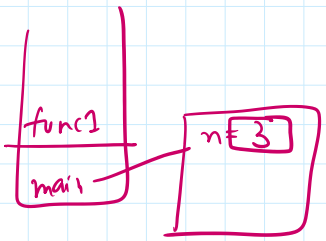
homogenous



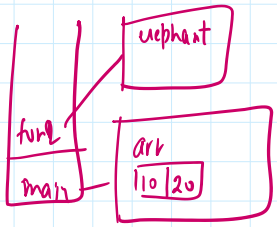
```
int func1(int elephant){
    elephant = 2;
}
```

pass by value

```
int n=3;
func1(n);
```



```
int fun2(int elephant[])
    elephant    = 1;
{
    int arr[2] = {10,20};
    func2(arr[0])
```



int elephant[] arr[0];

---

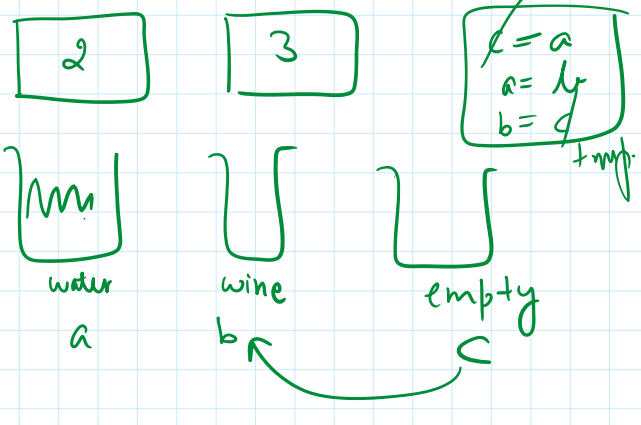
(2) a → myswap

(3) b → myswap

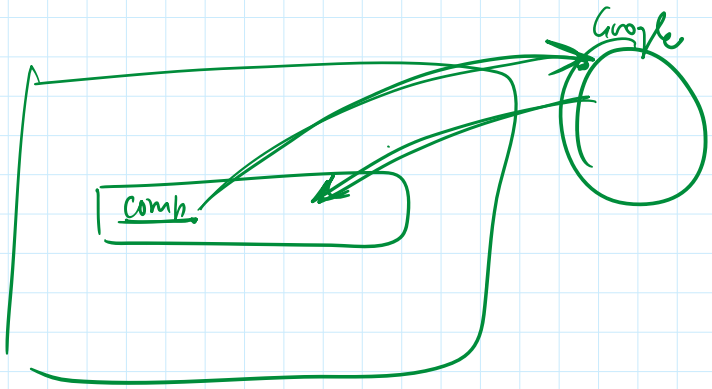
a = 3

b = 2

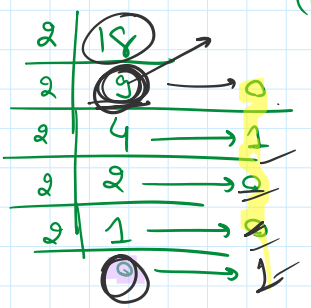
void



Google



$(18)_{10} \rightarrow (?)_2$




```
int dec2bin(int num)
{ int ans=0;
  int fact =1;
  while (num>0){
     int rem = num%2;
     int quo = num/2;
     ans = power*rem+ans
     power=power*10;
     num=quo;
  }
  return ans;}
```

0+1

10+0= 10

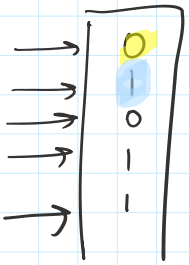
100*0 +10= 010

1000 *1+ 010 = 1010

0 1 0 0 1

1 0 1 0

(1) 1==0



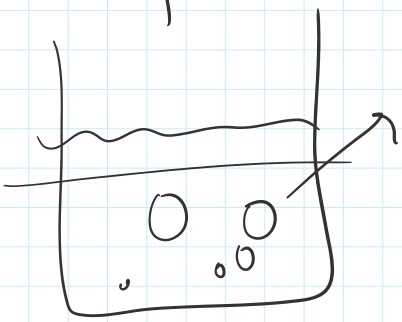
0 1 0 1 1

(1) 1 0 1 0 1 1 0 1 0

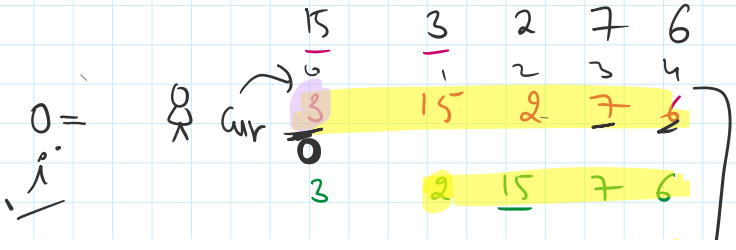
10^4 + 1010

power

Searching



Larger comes first



i

0

3 2 15 7 6

3 2 7 15 6

3 2 7 6 15

1

2 3 7 6 15

cur 2 3 7 6 15

2 3 6 7 15

2

2 3 1 7 15

2 1 3 7 15

3

1 2 3 7 15

$[0, n-1)$

$[0, n-1-i)$

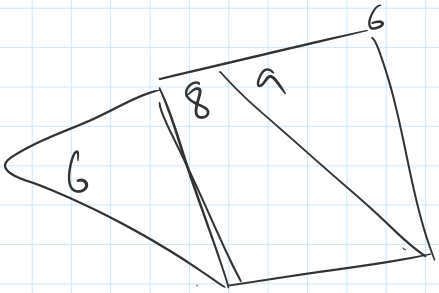
when $i=0$ 1 element is sorted in this iteration

$[n-1]$ n element

$i = [0, n-1]$

Insertion Sort

8



① To insert an element x into sorted range find an element smaller than x.

tmp = 1

10 8 3 4 1

3 8 10 4 1

3 4 8 10 1

2 3 — 5

j

Insertion Sort

④

```
for (int i=1; i<n; ++i)
    j=i-1;    int tmp = arr[i];
    while (j>=0 && arr[j] > arr[tmp]) {
```

```
j = i-1;    int tmp = arr[i];
while (j>=0 && arr[j] > ~~arr~~tmp) {
        arr[j+1] = arr[j]; --j
}
arr[j+1] = tmp;
}
```