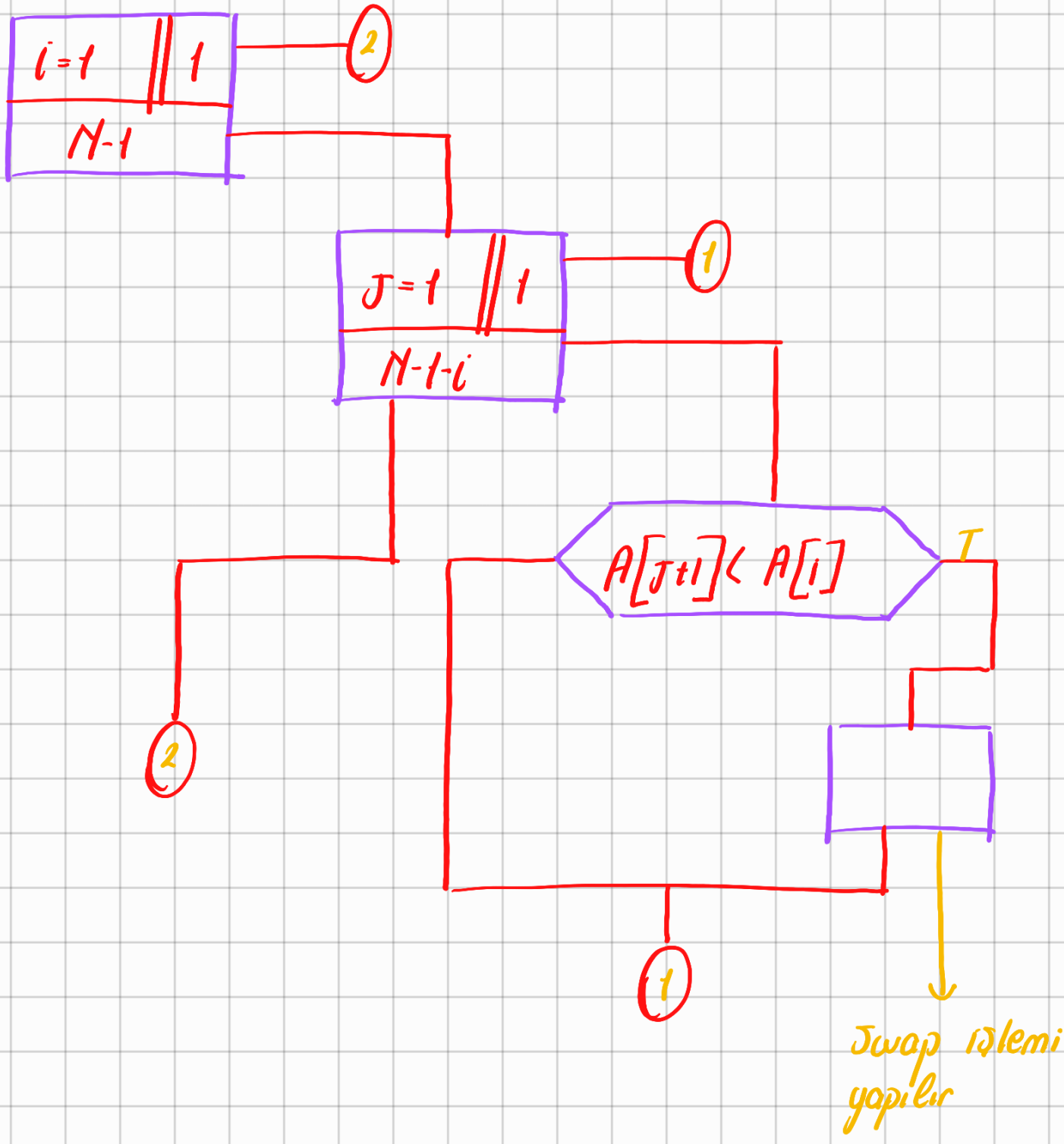
Bubble Sort



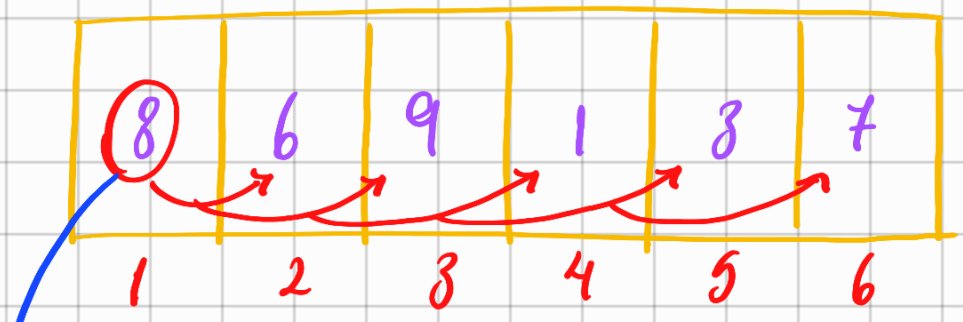
Kod

```
#include <stdio.h>
```

```
#define n 5
int main(){
 int a[n]={8,2,6,4,5};
 int i,j,tmp;
 for(i=0;i<n-1;i++)
  for(j=0;j<n-1-i;j++)
      if(a[j]>a[j+1]){
          tmp=a[j];
          a[j]=a[j+1];
          a[j+1]=tmp;
      }
  for(i=0;i<n;i++)
      printf("%d",a[i]);
          printf("\n");
 return 0;
}
```

Çıktı → 2, 4, 5, 6, 8

Selection Sort



→ ilk önce buna bakıyoruz
ve bütün diziyle sırasıyla
karşılaştırarak kaç tanesinden
büyük olduğuna bakarım 8 4 tane
elemandan büyük öğeyse 8'in yeri
5. indistir deriz

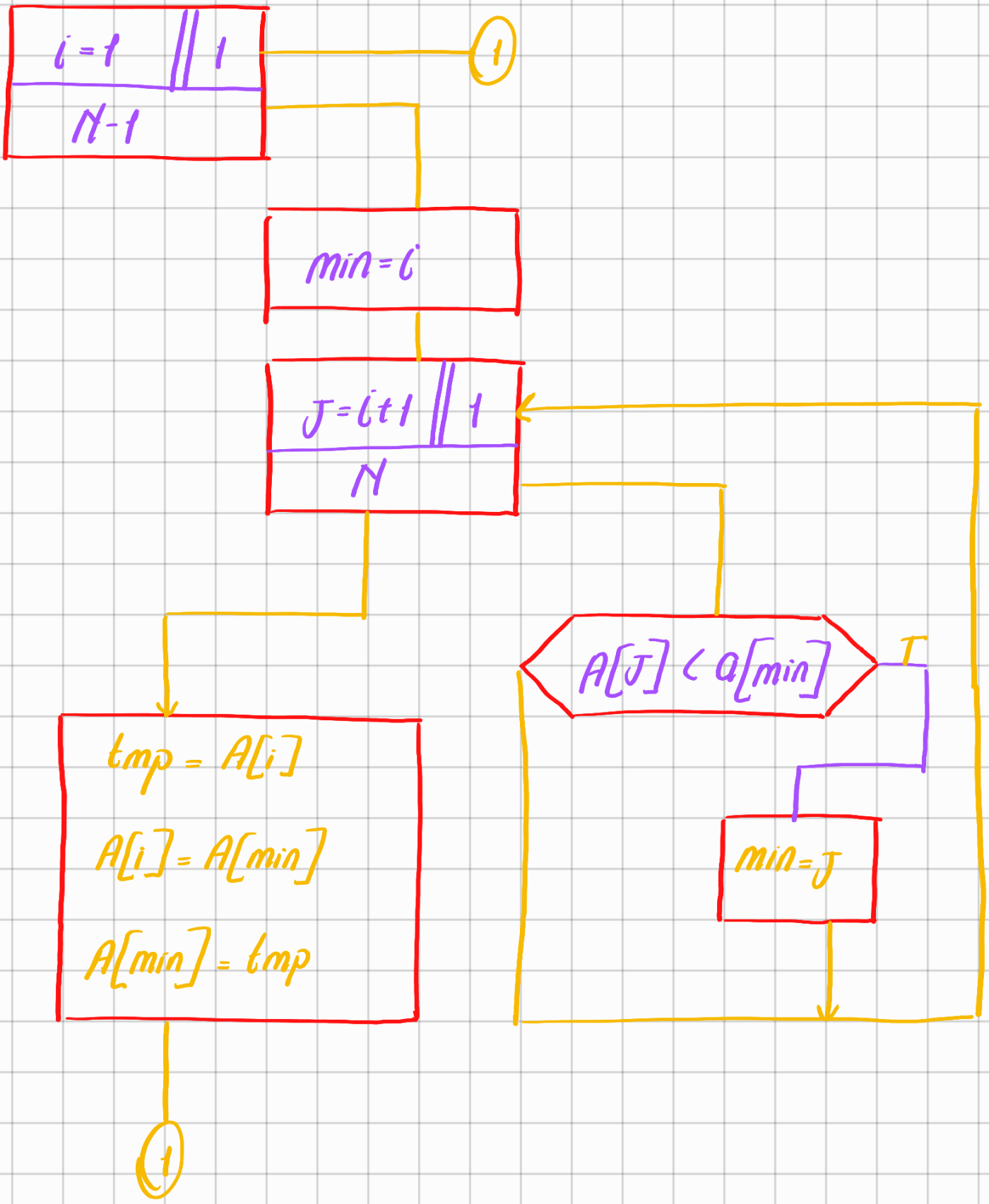
↓

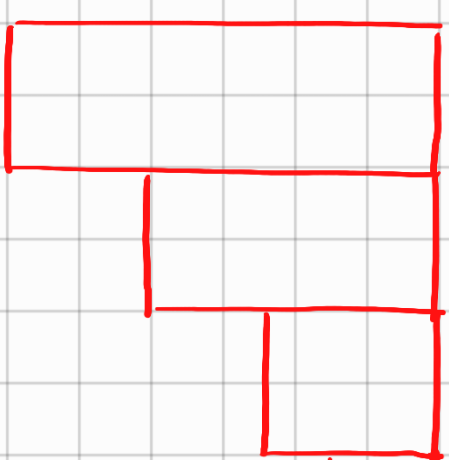
Bunu hepsi için yaparız

↓

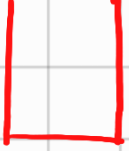
Bu bize ne kazandırdı
Swap operasyonu yapmadık

Algoritma





→ Bunu bir matris gibi düşünürsek matrisin yarısını tararız yani bubble sort-ta karmaşıklık N^2 iken burdaki karmaşıklık $N^2/2$ olur



Kaç yer değiştirme N olur

Selection Sort kodlama

```
#include <stdio.h>
#define n 5
int main(){
    int a[n]={9,1,5,2,8};
    int i,j,min,tmp;
    for(i=0;i<n-1;i++){
        min=i;
    for(j=i+1;j<n;j++){
        if(a[j]<a[min])
            min=j;
    }
    // swap
    tmp=a[min];
```

```
    a[min] = a[i];
    a[i] = tmp;
  }
  for(i=0; i<n; i++)
    printf("%d", a[i]);
  printf("\n");
return 0;
}
```

Insertion Sort algoritmasına bak