

1. พิจารณาอัลกอริทึม Quick Select ด้านล่าง

```
#include<stdio.h>
int arr[]={1, 5, 10, 4, 8, 2, 6, 9, 20};
int k=4;
int n =sizeof(arr)/sizeof(arr[0]);

int partition(int l, int r){
    ****use median of three for pivot selection
}

int quickSelect(int low, int high, int k){

    if(low ==high)
        return.....;

    int p =partition(arr, low, high);

    if(.....) //case k =Pivot position
        return .....;
    else if(.....) //case k ∈ L
        return quickSelect(arr, low, p-1, k);
    else { //case k ∈ R
        k =.....;
        return quickSelect(arr, ....., ....., .....);
    }
}

int main(){
    printf("%d", quickSelect(arr, ....., ....., k));
    return 0;
}
```

จงเติมโค้ดด้านบนเพื่อให้โปรแกรมทำงานได้สมบูรณ์ พร้อมแสดงผลลัพธ์ในแต่ละรอบของการทำงาน (หลังจากทำ partition)

2. จงเขียนโปรแกรมที่มีประสิทธิภาพ $O(\log n)$ เพื่อนับจำนวนข้อมูลซ้ำ เมื่อกำหนดชุดข้อมูลและ target มาให้ (assignment_4_1)
3. จงเขียนโปรแกรมที่มีประสิทธิภาพ $O(n)$ เพื่อค้นหาค่าเฉลี่ยของคะแนนสอบมากที่สุด K อันดับแรก (assignment_4_2)
4. จงแสดงขั้นตอนวิธีการคูณเลขจำนวนเต็มด้วยวิธี Karatsuba เมื่อกำหนดให้ $X = 123$, $Y = 345$

