typedef struct node{

    int posi;

    int speed;

}node;

int cmp (const void\*a , const void\*b){

    node aa=\*(node\*) a;

    node bb=\*(node\*) b;

    return bb.posi-aa.posi;

}

int carFleet(int target, int\* position, int positionSize, int\* speed, int speedSize){

    float time;//存取(target-position)/speed的數值

    float top=0;//存取目前stack中的最上面的車隊

    int index=0;//計算有多少車隊

    node \*newNode = malloc(sizeof(node)\*positionSize);

    for(int i=0; i<positionSize ;i++){

        newNode[i].posi=position[i];

        newNode[i].speed=speed[i];

    }

    qsort(newNode,positionSize,sizeof(node),cmp);//將所有車子依照出發位置按照大到小排列

    for(int i=0; i<positionSize ;i++){

        //printf("newNode[%d].posi=%d speed=%d\n",i,newNode[i].posi,newNode[i].speed);

        time = ((float)target-(float)newNode[i].posi)/(float)newNode[i].speed;//算出抵達目的地前所需要的時間

        //printf("acc=%f\n",acc);

        //printf("top=%f\n",top);

        //printf("index=%d\n",index);

        if( index==0 || top < time){//如果前面的車隊top所需要的時間time多，表示追得上，會在抵達目的地前加入車隊

            index++;

            top=time;

        }else{}

    }

    return index;

}