/\*\*

 \* Note: The returned array must be malloced, assume caller calls free().

 \*/

int\* dailyTemperatures(int\* temperatures, int temperaturesSize, int\* returnSize){

    int \*ans=malloc(sizeof(int)\*temperaturesSize);

    //printf("temperaturesSize=%d\n",temperaturesSize);

    int stack[temperaturesSize][2];

    int top=0 ;

    int topIndex=-1;

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

        //printf("%d \n",temperatures[i]);

        if( topIndex == -1){//如果topIndex為-1就直接push到stack中

            topIndex++;

            top=temperatures[i];

            stack[topIndex][0]=temperatures[i];

            stack[topIndex][1]=i;

        }else{

            if(temperatures[i] <=  top){//如果現在值temperatures[i]比top較小就將值放push到stack中，並且將top更換

                topIndex++;

                top=temperatures[i];

                stack[topIndex][0]=temperatures[i];

                stack[topIndex][1]=i;

            }else{

                while( temperatures[i] >  top){//比較stack中的top，如果比較大就將stack的top給pop出來並且將i-index放到答案中

                    //printf(" temperatures[i]=%d,top=%d,topIndex=%d\n",temperatures[i],top,topIndex);

                    ans[stack[topIndex][1]]=i-stack[topIndex][1];

                    topIndex--;

                    if(topIndex==-1){//如果topIndex--到清空stack就break

                        top=0;

                        break;

                    }else{

                        top=stack[topIndex][0];//如果還有值就替換stack最上面的值為top

                    }

                }

                //如果top上面的值沒有比較大就將現在值temperatures[i]push進去stack中

                topIndex++;

                top=temperatures[i];

                stack[topIndex][0]=temperatures[i];

                stack[topIndex][1]=i;

            }

        }

    }

    while( topIndex != -1){

        //printf("top=%d,topIndex=%d\n",top,topIndex);

        ans[stack[topIndex][1]]=0;

        top=stack[topIndex][0];

        topIndex--;

    }

    \* returnSize=temperaturesSize;

    return ans;

}