**int upper\_bound(int l,int h,int\* arr, int num)**

{

while(l<h)

{

int mid = (l+h)/2;

if(arr[mid]<=num)

{

l=mid +1;

}

else

{

h = mid;

}

}

return l;

}

**int lower\_bound(int l,int h,int\* arr, int num)**

{

while(l<h)

{

int mid = (l+h)/2;

if(arr[mid]>=num)

{

h=mid;

}

else

{

l = mid+1;

}

}

return l;

}

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

{

return (\*((int\*)a)- \*((int\*)b));

}

int\* jobOffers(int scores\_size, int\* scores, int lowerLimits\_size, int\* lowerLimits, int upperLimits\_size, int\* upperLimits, int\* result\_size) {

int i;

int \*res = (int\*)calloc(lowerLimits\_size,sizeof(int));

qsort(scores,scores\_size,sizeof(int),cmp);

for(i=0;i<lowerLimits\_size;i++)

{

int x = lower\_bound(0,scores\_size,scores,lowerLimits[i]);

int y = upper\_bound(0,scores\_size,scores,upperLimits[i]);

res[i] = y-x;

}

\*result\_size= lowerLimits\_size;

return res;

}