

int \*func(int a[], int b[])

{

int asize=0,bsize=0,combsize = 0,newsize=0;

//asize=size of a ,bsize=size of b,combsize=asize+bsize,newsize=size of the finished array

int i,j=0,temp=0;

for (i = 0; a[i] != -1; i++)//count the size of a

asize++;

asize++;

for (i = 0; b[i] != -1; i++)//count the size of b

bsize++;

bsize++;

combsize = asize + bsize;

int \*combarr = (int\*)malloc(sizeof(int)\*combsize);//הקצאה לכתובת למערך משולב משני המערכים

for (i = 0; i < combsize; i++)//combining the two arrays into one

{

if (i <= asize)//insert a into combarr

{

combarr[i] = a[j];

j++;

}

else if (i > asize)//insert b into combarr

{

combarr[i] = b[j];

j++;

}

if (combarr[i] == -1)//zero the counting befor inserting b

j = 0;

}

for (i = (combsize - 1); i >= 0; i--) //bubble sorting

for (j = 1; j <= i; j++) {

if (combarr[j - 1] > combarr[j]) {

temp = combarr[j - 1];

combarr[j - 1] = combarr[j];

combarr[j] = temp;

}

}

for (i = 1; i < combsize; i++)//count how many numbers there are without duplicates

{

if (combarr[i] != combarr[i - 1])

newsize++;

}

newsize++;//adds another slot

int \*newarr = (int\*)malloc(sizeof(int)\*newsize);//הקצאה דינאמית newarr

for (i = 1,j=0; i < combsize; i++)//inserting numbers without duplicates to newarr

{

if (combarr[i] != -1 && combarr[i]!=combarr[i-1])

{

newarr[j] = combarr[i];

j++;

}

}

for(i=0;i<newsize;i++)

if (j < newsize)

newarr[j] = -1;//adds -1 at the and of new arr

return newarr;

}

void main()

{

int a[] = { 2,3,6,1,1,1,2,4,5,2,2,2,-1};

int b[] = { 1,6,2,5,4,2,3,6,9,5,5,-1};

func(a, b);

}