```
//Name: Mehmet Fatih Çelik
//Id: 2385268

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int main(){
	srand(time(NULL));
	int size, *firstArray, *secondArray, *yourArray, i;
	double firstDistance = 0, secondDistance = 0;

	printf("Enter the array size: ");
	scanf("%d",&size);

	firstArray = (int *)malloc(sizeof(int)*size);
	secondArray = (int *)malloc(sizeof(int)*size);
	yourArray = (int *)malloc(sizeof(int)*size);

	if (firstArray==NULL || secondArray==NULL || yourArray==NULL){
		printf("Error while allocating the memory!");
		exit(-1);
	}

	printf("\nFirst array created as: ");
	for(i=0;i<size;i++){ // populates both arrays at the same time prints the firstArray, I think this way is more effecient.
		firstArray[i] = rand()%101;
		printf("%d ",firstArray[i]);
		secondArray[i] = rand()%101;
	}
```

```
printf("\nSecond array created as: ");
for(i=0;i<size;i++) // prints the secondArray
	printf("%d ",secondArray[i]);

printf("\n\nEnter your array data: ");
for(i=0;i<size;i++){
	if (i == size-1){ // for last element of yourArray
		scanf("%d",&yourArray[i]);
		continue;
	}
	scanf("%d ",&yourArray[i]);
}

for(i=0;i<size;i++){
	firstDistance += pow(firstArray[i]-yourArray[i], 2);
	secondDistance += pow(secondArray[i]-yourArray[i], 2);

}
firstDistance = sqrt(firstDistance);
secondDistance = sqrt(secondDistance);

printf("\nDistance of your array to the first array: %.4lf",firstDistance);
printf("\nDistance of your array to the second array: %.4lf",secondDistance);

if(firstDistance > secondDistance)
	printf("\n\nYour array is more similar to the second array!");
else if(firstDistance < secondDistance)
	printf("\n\nYour array is more similar to the first array!");
else
	printf("\n\nYour array is the same to the both arrays!");
```

```
        free(firstArray);
        free(secondArray);
        free(yourArray);

        return 0;
}
```