# GTU Department of Computer Engineering
## CSE 331 - FALL 2020
## COMPUTER ORGANIZATION HW2

**161044063**

**ZAFER ALTAY**

# 1-)CODE EXPLANATIONS

First of all, I removed some of the optimizations I made from the c code because I got errors while converting to assembly.So my homework is missing optimizations and bonus parts.But others other than that are working perfectly, if your tests do not show any errors.

**Now let me make some explanations in C code:**

First of all, I did not change the main function. I just added the following line. Since the variable sizeCounter in this line is a variable that I use globally, I had to assign in the main.I did this so that the recursion function reads the array from last to first while backtracking.

```
sizeCounter=arraySize-1;
```

**Now let me explain my recursion function:**

These are my base cases.Num: Subtracting the number that I am checking now from target and if the remaining number equals 0, it means we have reached the number we are looking for.If num less than 0,it tells us that the number we looked at is exceeding.The last base case is checking whether we exceed array size.

```
if(num==0) return 1;
if(num<0) return 0;
if(sizeCounter-size+1==sizeCounter+1) return 0;
```

Here I call recursion continuously and go from the last to the first element of the array, if there is one that matches the target number directly, the function ends immediately.

```
int val1 = CheckSumPossibility(num,arr,size-1);
if(val1==1){
return 1;}
```

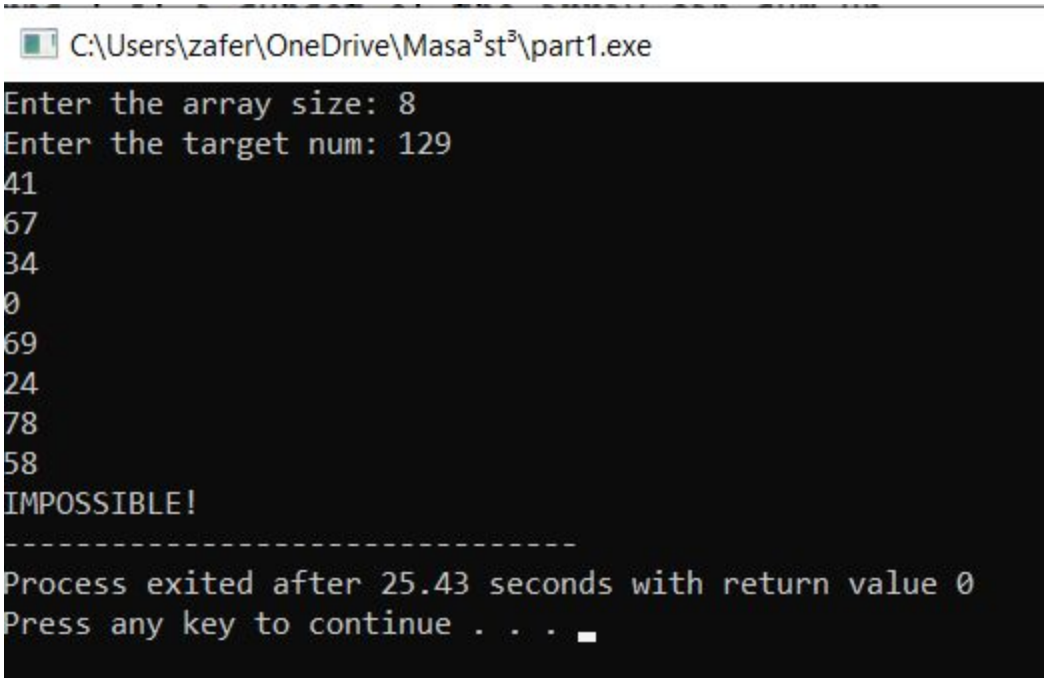
If the function is not finished until the first element, this time I see if I can reach the target by starting from the first element and collecting it backwards from where I am in each step and when I'm sending this for the next element, I'm sending it by removing the element I was on, which means I'm adding it.If my add less or equals 0, I return the result according to what comes from the next function.

**NOTE:**In the code, as in main, it wants first the size and then the target number. It wants input, number of sizes times. After entering each number, press enter.It is same way in two codes.

# 2-)RESULT ANALYSIS

I tested the conditions in the pdf here.

## 1.TEST:




```
Enter the array size8
Enter the target num129
41
67
34
0
69
24
78
58
Impossible!
-- program is finished running --
```

## 2.TEST:

```
Enter the array size: 8
Enter the target num: 129
62
64
5
45
81
27
61
91
IMPOSSIBLE!
--------------------------------
Process exited after 18.81 seconds with return value 0
Press any key to continue . . .
```

```
Enter the array size8
Enter the target num129
62
64
5
45
81
27
61
91
Impossible!
-- program is finished running --
```

## 3.TEST:

```
Enter the array size: 8
Enter the target num: 129
95
42
27
36
91
4
2
53
POSSIBLE!
--------------------------------
Process exited after 25.04 seconds with return value 0
Press any key to continue . . .
```

```
Enter the array size8
Enter the target num129
95
42
27
36
91
4
2
53
Possible!
-- program is finished running --
```

## 4.TEST:

```
Enter the array size: 8
Enter the target num: 129
92
82
21
16
18
95
47
26
POSSIBLE!
--------------------------------
Process exited after 19.04 seconds with return value 0
Press any key to continue . . .
```

```
Enter the array size8
Enter the target num129
92
82
21
16
18
95
47
26
Possible!
-- program is finished running --
```

## 5.TEST:

```
Enter the array size: 8
Enter the target num: 129
71
38
69
12
67
99
35
94
POSSIBLE!
--------------------------------
Process exited after 17.27 seconds with return value 0
Press any key to continue . . .
```

```
Enter the array size8
Enter the target num129
71
38
69
12
67
99
35
94
Possible!
-- program is finished running --
```

## 6.TEST:

```
Enter the array size: 8
Enter the target num: 129
3
11
22
33
73
64
47
11
IMPOSSIBLE!
--------------------------------
Process exited after 17.64 seconds with return value 0
Press any key to continue . . .
```

```
Enter the array size8
Enter the target num129
3
11
22
33
73
64
47
11
Impossible!
-- program is finished running --
```