

### Refik Orkun Arslan

```
bool CheckSumPossibility(int arr[], int n, int sum)
{
```

```
    if (sum == 0)
```

your program should stop when only one possible combination is found.

```
    {
```

```
        return true;
```

```
    }
```

```
    if (n == 0)
```

```
    {
```

```
        return false;
```

```
    }
```

```
    if (arr[n - 1] > sum)
```

```
    {
```

```
        return CheckSumPossibility(arr, n - 1, sum);
```

 . you must ignore the next recursive calls when the sum exceeded the target number num.

```
    }
```

```
    return CheckSumPossibility(arr, n - 1, sum)
```

```
        || CheckSumPossibility(arr, n - 1, sum - arr[n - 1]);
```

```
}
```

CheckSumPossibility:

```
    addi $sp, $sp, -16
```

```
    addi $t0, $a0, -1
```

```
    sw  $t0, 0($sp)
```

```
    sw  $ra, 4($sp)
```

```
    sw  $t6, 12($sp)
```

```
    bne $a0, $zero, ELSE2
```

```
    li  $v0, 0
```

```
    addi $sp, $sp, 16
```

```
    jr  $ra
```

```
    if (sum == 0)
```

```
    {
```

```
        return true;
```

```
    }
```

ELSE2:

```
    bne $t6, $zero, ELSE3
```

```
    li  $v0, 1
```

```
    addi $sp, $sp, 16
```

```
    li  $v0, 4
```

```
    la  $a0, string
```

```
    syscall
```

```
    jr  $ra
```

```
    else if (n == 0)
```

```
        return false;
```

ELSE3:

```
    sll $t3, $t0, 2
```

```
    add $t0, $t0, $t3
```

```
    blt $t2, $t6, ELSE4
```

```
    else if (arr[n - 1] > sum)
```

```
    {
```

```

    sw $t2, 8($sp)
move $a1, $t0
jal CheckSumPossibility
ELSE4:
    addi $a0,$a0,-1
    move $a1, $t0
    jal CheckSumPossibility

    add $t1, $t1,$a1
    lw $t2, 0($t1)
    sub $t6,$t6,$t2
    or $v0, $v0, $t6
    lw $ra, 4($sp)
    addi $sp, $sp, 16
    jr $ra

```

```

    return CheckSumPossibility(arr, n - 1, sum);
}

else

    return CheckSumPossibility(arr, n - 1, sum)
|| CheckSumPossibility(arr, n - 1, sum - arr[n - 1]);

```

8 129  
41 67 34 0 69 24 78 58  
Not possible!

```

8
129
41
67
34
0
69
24
78
58
Not possible!

```

8 129  
62 64 5 45 81 27 61 91  
Not possible!

```

8
129
62
64
5
45
81
27
61
91
Not possible!

```

8 129  
95 42 27 36 91 4 2 53  
Possible!

```

8
129
95
42
27
36
91
4
2
53
Possible!

```

8 129  
92 82 21 16 18 95 47 26  
Possible!

```

8
129
92
82
21
16
18
95
47
26
Possible!

```

8 129  
71 38 69 12 67 99 35 94  
Possible!

```

8
129
71
38
69
12
67
99
35
94
Possible!

```

8 129  
3 11 22 33 73 64 41 11  
Not possible!

```

8
129
3
11
22
33
73
64
41
11
Not possible!

```

8 129 41 67 34 0 69 24 78 58 Not possible!

\$zero	0	0x00000000
\$at	1	0x10010000
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x1001002e
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0xffffffff
\$t1	9	0x10010000
\$t2	10	0x00000029

Not possible

a0=sum

If a0 is zero, possible;

-- program is finished running --

Because of CheckSumPossibility(arr, n - 1, sum - arr[n - 1]);  
62 64 5 45 81 27 61 91

\$zero	0	0x00000000
\$at	1	0x10010000
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x1001002e
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x10010000
\$t2	10	0x0000003e
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0xffffffffe91

Not possible

8 129 62 64 5 45 81 27 61 91 Not possible!

-- program is finished running --

\$zero	0	0x00000000
\$at	1	0x00000001
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x10010000
\$t2	10	0x0000005f
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0xffffffffd89

Possible

-- program is finished running --

\$zero	0	0x00000000
\$at	1	0x10010000
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x10010024
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x10010000
\$t2	10	0x0000005e
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0xffffffffda1

Possible

-- program is finished running --

\$zero	0	0x00000000
\$at	1	0x10010000
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x10010024
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x10010000
\$t2	10	0x00000047
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0xffffffffe49

Possible

-- program is finished running --

\$zero	0	0x00000000
\$at	1	0x10010000
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x10010024
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x10010000
\$t2	10	0x00000003
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000069

Not possible

-- program is finished running --