CSE 331 – Computer Organization

Homework #2 Report

Abdullah Çelik

171044002

Procedures Overview

- getNonNegIntWithText

  Print message and gets non-negative integer. If user enters negative number, prints error message and again ask user a non-negative integer.

  Arguments and Outputs:

  $a0 = address of text

  $a1 = address of error text

  $v0 = non-negative integer
- printArray

  Prints given array using given size on the screen.

  Arguments:

  $a0 = address of array

  $a1 = size of array
- checkSumPossibility

  Checks whether if a subset of array elements can sum up to the target number.

  Arguments and Outputs:

  $a0 = target number

  $a1 = address of array

  $a1 = size of array

  $v0 = 0 or 1. If 0, possible sub array was not found. Otherwise, found.
- printResult

  Prints “Possible!” if given parameter is 0. Otherwise, “Not Possible!”.

  Arguments:

  $a0 = state
- initializeVector

  The vector contains the array address, size, and capacity. Creates a dynamic vector for the given address. If the given capacity is a non-positive integer, it sets its capacity to 1. If the capacity is positive, it creates a vector at the given capacity.

  Arguments:

  $a0 = address of vector

  $a1 = capacity
- addItemToVector

  Adds the given item to the given vector. If size reaches capacity, it does the reallocation process itself.

  Arguments:

  $a0 = address of vector

  $a1 = item
- fillVector

  Takes integer from user and adds it to given vector. This process continues until given size.

  Arguments:

  $a0 = address of vector

  $a1 = size

- freeMemory

  Returns given byte from the given dynamic address to the heap.

  Arguments:
  $a0 = address of block memory
  $a1 = number of words to deallocate

Improvements

The actions to be taken by giving messages to the user were shown.

The user was prevented from entering negative numbers. As long as user enters a negative number, it is expected to receive an error message and enter an integer again.

By creating a vector structure

- The array size that the user can enter is not restricted because space is dynamically allocated. User can create an array of desired size.
- It is easier to add items received from the user to the vector. In this way, it became easier to get a number of elements from the user.
- The sub-array elements found by the checkSumPossibility procedure are filled into a new vector instead of printing on the screen. Then new vector is printed.
- The received dynamic memory can be returned to heap again.

Sub-arrays exceeding the target number were not created in checSumPossibility, and precodure was terminated after the first sub-array found. In this way, the performance has been improved.

Test Cases

Assembly:

```
Size: 8
Target number: 129
1. element: 41
2. element: 67
3. element: 34
4. element: 0
5. element: 69
6. element: 24
7. element: 78
8. element: 58
Not Possible!
-- program is finished running --
```

```
Size: 8
Target number: 129
1. element: 62
2. element: 64
3. element: 5
4. element: 45
5. element: 81
6. element: 27
7. element: 61
8. element: 91
Not Possible!
-- program is finished running --
```

```
Size: 8
Target number: 129
1. element: 95
2. element: 42
3. element: 27
4. element: 36
5. element: 91
6. element: 4
7. element: 2
8. element: 53
Possible!(2 91 36 )
-- program is finished running --
```

```
Size: 8
Target number: 129
1. element: 92
2. element: 82
3. element: 21
4. element: 16
5. element: 18
6. element: 95
7. element: 47
8. element: 26
Possible!(16 21 92 )
-- program is finished running --
```

```
Size: 8
Target number: 129
1. element: 3
2. element: 11
3. element: 22
4. element: 33
5. element: 73
6. element: 64
7. element: 41
8. element: 11
Not Possible!
-- program is finished running --
```

```
Size: 8
Target number: 129
1. element: 71
2. element: 38
3. element: 69
4. element: 12
5. element: 67
6. element: 99
7. element: 35
8. element: 94
Possible!(94 35 )
-- program is finished running --
```

```
Size: -5
Number should be non-negative! Please enter non-negative integer: -10
Number should be non-negative! Please enter non-negative integer: 10
Target number: -129
Number should be non-negative! Please enter non-negative integer: 129
1. element: -123
Number should be non-negative! Please enter non-negative integer: 123
2. element: 0
3. element: -3
Number should be non-negative! Please enter non-negative integer: 3
4. element: 4
5. element: 5
6. element: 6
7. element: 7
8. element: 8
9. element: 9
10. element: 10
Possible!(6 0 123 )
-- program is finished running --
```

C++:

```
Size: 8
Target number: 129
1. element: 41
2. element: 67
3. element: 34
4. element: 0
5. element: 69
6. element: 24
7. element: 78
8. element: 58
Not possible!
```

```
Size: 8
Target number: 129
1. element: 62
2. element: 64
3. element: 5
4. element: 45
5. element: 81
6. element: 27
7. element: 61
8. element: 91
Not possible!
```

```
Size: 8
Target number: 129
1. element: 95
2. element: 42
3. element: 27
4. element: 36
5. element: 91
6. element: 4
7. element: 2
8. element: 53
Possible!(2 91 36 )
```

```
Size: 8
Target number: 129
1. element: 92
2. element: 82
3. element: 21
4. element: 16
5. element: 18
6. element: 95
7. element: 47
8. element: 26
Possible!(16 21 92 )
```

```
Size: 8
Target number: 129
1. element: 3
2. element: 11
3. element: 22
4. element: 33
5. element: 73
6. element: 64
7. element: 41
8. element: 11
Not possible!
```

```
Size: 8
Target number: 129
1. element: 71
2. element: 38
3. element: 69
4. element: 12
5. element: 67
6. element: 99
7. element: 35
8. element: 94
Possible!(94 35 )
```

```
Size: -5
Number should be non-negative! Please enter non-negative integer: -10
Number should be non-negative! Please enter non-negative integer: 10
Target number: -129
Number should be non-negative! Please enter non-negative integer: 129
1. element: -5
Number should be non-negative! Please enter non-negative integer: 5
2. element: -15
Number should be non-negative! Please enter non-negative integer: 15
3. element: 5
4. element: 4
5. element: 3
6. element: 2
7. element: 9
8. element: 10
9. element: 11
10. element: 12
Not possible!
```