CSC209H Worksheet: Stacks and Heaps

1. Trace the memory usage for the program below. We have set up both stack frames for you, and the location of the heap.

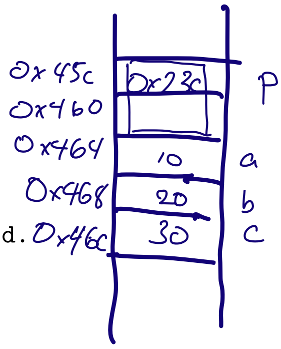
for Q2 →

Section	Address	Value	Label
Heap	0x23c	10	
	0x240	20	
	0x244	30	
	0x248		
	⋮	⋮	
stack frame for mkarray	0x454		
	0x458		
	0x45c	0x474	p
	0x460		
	0x464	10	a
	0x46c	20	b
	0x470	30	c
	0x474	10	arr
	0x478	20	
	0x47c	30	
stack frame for main	0x480	0x474	ptr
	0x484		
	0x488		
	0x48c		

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

int *mkarray(int a, int b, int c) {
    int arr[3];
    arr[0] = a;
    arr[1] = b;
    arr[2] = c;

    int *p = arr;
    return p;
}

// Code for other_function() omitted.

int main() {
    int *ptr = mkarray(10, 20, 30);
    other_function();
    printf("%d %d %d\n", ptr[0], ptr[1], ptr[2]);
}
```

Q2 ↓

→ int * p= malloc(3 * sizeof(int))
p[0]=a;
p[1]=b;
p[2]=c;
return p;



other_function will probably overwrite the memory used by local variables in mkarray

2. The program in part 1 will not work correctly. Notice the call to other_function. Explain to your partner why the program doesn't work. Fix the mkarray function, and trace it again.

3. Once you've fixed the code, add a statement to your program to deallocate the memory on the heap as soon as possible.

free must be after the print statement

CSC209H Worksheet: Stacks and Heaps

4. Trace the memory usage for the program below. We have set up the stack frames for you, and the location of the heap.

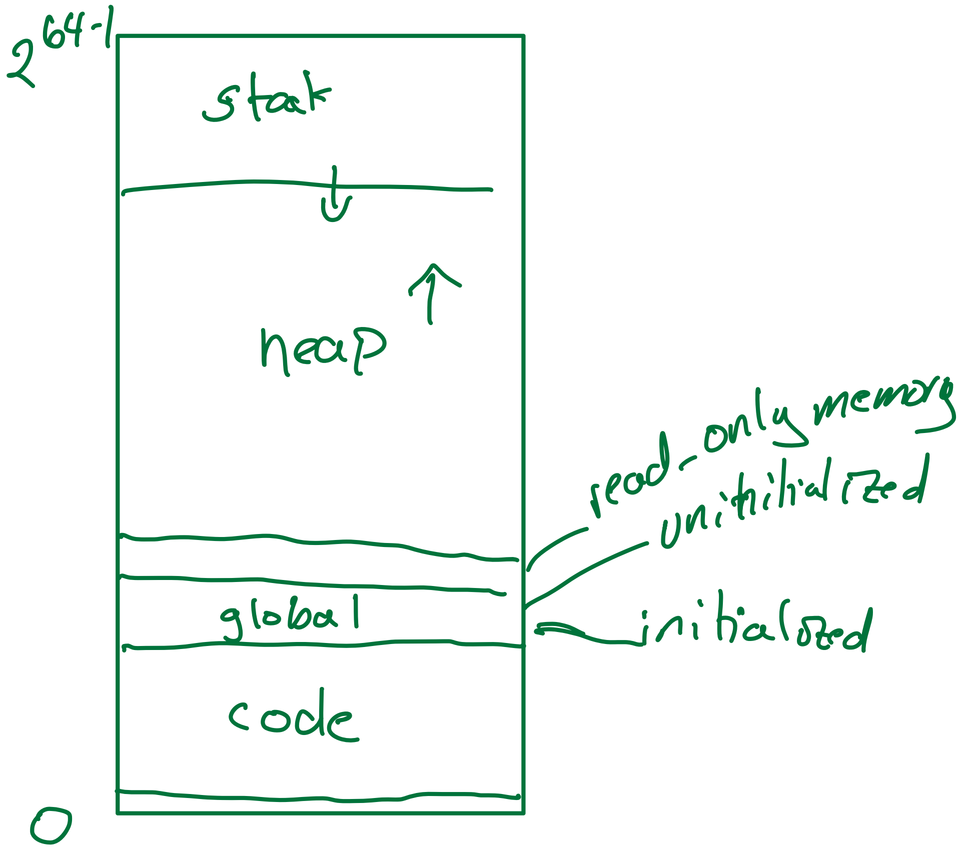
```
#include <stdio.h>
#include <stdlib.h>

/* Build an array in dynamic memory to hold
   multiples of x from x to x*x.
   Return a pointer to this array.
*/
int *multiples(int x) {
    int *a = malloc(sizeof(int) * x);
    for (int i = 0; i < x; i++) {
        a[i] = (i + 1) * x;
    }
    return a;
}

int main() {
    int *ptr;
    int size = 3;

    ptr = multiples(size);

    for (int i = 0; i < size; i++) {
        printf("%d\t", ptr[i]);
    }
    printf("\n");

    return 0;
}
```

Section	Address	Value	Label
Heap	0x224		
	0x228	3	
	0x22c	6	
	0x230	9	
	0x234		
	0x238		
	0x23c		
	0x240		
	0x244		
	⋮	⋮	
stack frame for multiples	0x46c	~~0x228~~	a
	0x470		
	0x474		i
	0x478	~~3~~	x
stack frame for main	0x47c	0x228	ptr
	0x480		
	0x484	3	size
	0x488		i
	0x48c		

5. Change the `main` function so that it calls `multiples` and prints the array in a loop with sizes of 3, 4, and 5. Besides the changes described, do not make any other changes or additions to the code.

6. Trace the memory usage of your changed program. Explain the problem to your partner and then fix it by adding calls to deallocate the memory.

Address Space


$2^{64}-1$
stack
heap
read_only memory
uninitialized
global
initialized
code
0