Memory Layout and Management Part 1

JinYoung Park [001064438]

Question 1a

When the two collide, data stored in one region may overwrite data stored in the other region. This overlap can corrupt the data in both regions, rendering the program's state invalid. As a consequence, the program might crash immediately, often resulting in a segmentation fault.

Question 1b

If a program directly calls operating system function addresses, it may fail if the operating system is updated or the function addresses change, as the program would reference outdated or incorrect addresses. However, using a system call table ensures that even if the function addresses change, the table entries can be updated, allowing the program to continue functioning correctly.

Question 1c

The `calloc` function requires two arguments: the number of elements to allocate and the size of each element.

```
char* block = calloc(2048, sizeof(char));
```

The `&block` as the first argument to `memset`, which is the address of the pointer `block`, not the memory block it points to. This results in overwriting the pointer.

```
memset(block, 0xFF, 2048);
```

```
int main() {
    char* block = calloc(2048, sizeof(char));
    if (block == NULL) {
        exit(EXIT_FAILURE);
    }
    memset(block, 0xFF, 2048);
    free(block);
    return 0;
}
```