

CS 4400 - Problem Set 5

Rob Johansen

u0531837

1. Problem 3.63. My reverse engineering skills lead me to conclude that E1 and E2 are defined as follows:

```
#define E1(n) (2n + 1)
#define E2(n) (3n)
```

2. Problem 3.64:

- A. The value on the stack at 8(%ebp) is effectively a pointer to an address on the stack that serves as an offset for where the sum and difference will eventually be stored.

The value on the stack at 12(%ebp) is the value of the variable x passed to the prod() function (also the value of s1.a).

The value on the stack at 16(%ebp) is the address of the variable y (&y) passed to the prod function (also the value of s1.p).

- B. The lowest field, at 0(%esp), is effectively a pointer to an address on the stack that serves as an offset for where the sum and difference will eventually be stored.

The next highest field, at 4(%esp), is the value of the variable x passed to the prod() function (also the value of s1.a).

The next highest field, at 8(%esp), is the address of the variable y (&y) passed to the prod() function (also the value of s1.p).

The next highest field, at 12(%esp), get used as the value of s2.sum.

The next highest field, at 16(%esp), get used as the value of s2.diff.

- C. The general strategy for passing a structure as an argument to a function is to place the members of the structure on the stack in the opposite order in which those objects were declared inside the

structure.

- D. The general strategy for handling a structure as a return value from a function is for the caller to allocate space on the stack in its own frame for the members of the structure, and for the callee to store the value of each member on the stack in the caller's frame.

3. Problem 3.65. My reverse engineering skills lead me to conclude that the values of A and B are defined as follows:

```
#define A 18
#define B 10
```

4. Problem 3.66. My reverse engineering skills lead me to conclude that:

A. The value of CNT is 2.

B. The complete declaration of a_struct looks like this:

```
typedef struct {
    int x[22];
    int idx;
} a_struct;
```