CSC 2400 – Understanding the X86 Stack Layout

The following exercises must be completed on **felix** or **helix**.

Exercise 1.
Copy the executable **x86stack1** from the **/tmp** directory into your **csc2400/X86** directory. Invoke **gdb** with **x86stack1** as an argument:

```
gdb ./x86stack1
```

Disassemble the **main** function (intel format), then answer the following questions:

a) Name the two functions invoked by the main functions:

b) How many arguments does the 2nd invoked function take? To answer this, you need to check what gets placed onto the stack just before the function is invoked with **call**. Write down the instructions that place the function arguments onto the stack.

c) What are the values of the arguments passed to the 2nd function invoked by **main**? There are two ways to answer this question (do it ***both*** ways, please – the result should be the same):

c.1 Set a breakpoint at the second function call (identified in part a)

```
break *(main+???)
```

Check the values that just got placed onto the stack

```
x /d $esp
x /d $esp+4
```

c.2 Disassemble the 2nd function invoked by **main**
Set a breakpoint at the first instruction following the common prolog

```
push ebp
mov  ebp, esp
```

Check the values stored at addresses **$ebp+8, $ebp+12**, etc

```
x /d $ebp+8
x /d $ebp+12
```

d) What is the value returned by the 2nd function invoked by **main**? (Check the contents of **eax** before the function returns.)

e) What is the value returned by **main**?

Exercise 2 [Repeat the activities of Exercise 1 on a different piece of code]:

Copy the executable **`x86stack2`** from the **`/tmp`** directory into your **`csc2400/X86`** directory. Disassemble the **`main`** function, then answer the following questions:

a) Name the two functions invoked by the main functions:

b) How many arguments does the 2nd invoked function take? To answer this, you need to check what gets placed onto the stack just before the function is invoked with **`call`**. Write down the instructions that place the function arguments onto the stack.

c) What are the values of the arguments passed to the 2nd function invoked by **`main`**?

d) What is the value returned by the 2nd function invoked by **`main`**?

e) What is the value returned by **`main`**?