Array Overflowing

This is a **pair** exercise and must be competed in your **tutorial** or **lab** with your partner.

Download `arrayOverflowing.c`, or copy it into your current directory on a CSE system by running

```
$ cp /web/cs1511/17s2/week07/files/arrayOverflowing.c .
```

In lectures we anticipated what would happen when you tried to inspect values of the index outside the array. Now try it out. Modify the code below to include a while loop so you can repeatedly test value without having to run the program multiple times.

Tip: You will need to use `gcc` for this activity, not `dcc`!

- Can you find where the canary variables are stored?
- Can you work out where and how gcc stores information in the frame for this function?
- [Does/How much does] the location of variables change if you run your program multiple times?
- What is the effect on the location of the variables of using/not using the -O flag when compiling with gcc?
- What else can you discover?

Start the activity - keep detailed lab notes about what you do, you code, what you find, and your thoughts. Submit this by creating a blog post.

Note: you may wish to write your notes and findings as you go in a text document using `gedit`, but make sure you post your findings in your blog when you're finished.

```
int canaryA = 42;
int testArray[10] = {0,1,2,3,4,5,6,7,8,9};
int canaryB = 99;

int index;
printf ("Enter an array index between 0 and 9\n");
scanf ("%d", &index);
printf ("testArray[%d]=%d,\n",index, testArray[index]);
```

To run Styl-o-matic:

```
$ 1511 stylomatic arrayOverflowing.c
Looks good!
```

You'll get advice if you need to make changes to your code.

Submit your work with the *give* command, like so:

```
$ give cs1511 wk07_arrayOverflowing
```

Or, if you are working from home, upload the relevant file(s) to the `wk07_arrayOverflowing` activity on Give Online.