CS 35L Software Construction Laboratory (Lab8-B)
Wed, Feb 29, 2012, Ver 1.1

Another Example on Buffer Overflow

```
   +-------------------------------------------+
    void function(int a, int b, int c) {
      char buffer1[5];
      char buffer2[10];
      int *ret;

      ret = buffer1 + 12;
      (*ret) += 8;
    }

    void main() {
      int x;

      x = 0;
      function(1,2,3);
      x = 1;
      printf("%d\n",x);
    }
  +-------------------------------------------+
```

```
bottom of                                                        top of
memory                                                           memory
           buffer2       buffer1   sfp   ret   a     b     c
<------   [           ][       ][    ][    ][    ][    ][    ]

top of                                                         bottom of
stack                                                              stack
```

  Note: memory can only be addressed in multiples of the word size.
        A word in this example is 4 bytes, or 32 bits.

Makefile and GCC Revisited

    In Lab 7, you are required to use GCC with several options, like "-S",
  "-fno-stack-protector". You are not going to run gcc command directly.
    Instead, you can achieve this by modifying the Makefile, especially the
  "$CFLAGS" variable.

```
   +-------------------------------------------+
    CC     = gcc
    CFLAGS = -g -fno-stack-protector

    all: helloworld

    helloworld: helloworld.o
            # Commands start with TAB not spaces
            $(CC) $(LDFLAGS) -o $@ $^

    helloworld.o: helloworld.c
```

```
            $(CC) $(CFLAGS) -c -o $@ $<

    clean:
            rm -f helloworld helloworld.o
   +-----------------------------------------+
```

```
  "-S" option of gcc
    Stop after the stage of compilation proper; do not assemble.
    The output is in the form of an assembler code file for each non-assembler
  input file specified.

GDB Revisited
  -- Attach GDB to a process which is running:
     $ gdb
     $ attach PID

Setup Environment for mudflap
  (Thanks to Jihyoung "Joseph" Kim for sharing his notes on mudflap)

  for Ubuntu
  $ sudo apt-get install gcc-opt
  $ sudo apt-get install libgcc1
  $ sudo apt-get install libgcc1-dbg
  $ sudo apt-get install libmudflap0
  $ sudo apt-get install libmudflap0-dbg
  $ sudo apt-get install libmudflap0-4.5-dev

  on SEAS lab lnxsrv
  $ bash
  $ export PATH='/usr/local_cs/linux/bin'
  $ export LD_LIBRARY_PATH=/usr/local_cs/linux/lib

  available port for section 1: 12101-12130

More about Lab 8
  -- Make sure your thttpd is working:
     1) Try visit http://localhost:80   Note: the port may not be 80
     2) Find help with commands: "ps" and "grep"
```