CS360 Midterm 1– Spring, 2016 – James S. Plank – February 16

In all of these questions, please assume the following:

- Pointers and longs are 4 bytes.
- The machine is little endian (like our lab machines and my Mac). So, if an the integer is 0xabcdef88, then its first byte is 0x88, and its last byte is 0xab.
- If you print the null character with %c, it will print nothing.
- There are no segmentation violations or bus errors in any of this code.

Question 1:

What is the output of this program?

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

main()
{
  char s[50];
  char *x, *y;

  strcpy(s, "ABCDEFGHIJKLMNOPQRSTUVWXYZ");
  x = s + 2;
  y = x + 5;
  strcpy(x, "01234567");
  strcat(y, "abcde");
  printf("%s\n", s);
  printf("%s\n", x);
  printf("%s\n", y);
  printf("%s\n", x+15);
}
```

Question 2:

The first line of this program's output is:

```
0x30 0x41 0x61 0x0
```

What is the rest of the output?

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

main()
{
  char s[50];
  char *argv[3];
  int *ip;

  printf("0x%x 0x%x 0x%x 0x%x\n",
          '0', 'A', 'a', '\0');

  argv[0] = "a.out";
  argv[1] = "abc";
  argv[2] = "AB";

  strcpy(s, "012345678901234567890");
  memcpy(s+12, argv[1], strlen(argv[1]));
  printf("%s\n", s);

  ip = (int *) s;
  memcpy(ip+1, argv[2], strlen(argv[2])+1);
  printf("%s\n", s);
  printf("0x%x\n", ip[2]);
}
```

Question 3:

What is the output of this program?

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct {
  int i;
  char c1;
  char c2;
} XX;

main()
{
  char s[50];
  XX *x;
  int j;

  strcpy(s, "ABCDEFGHIJKLMNOPQRSTUVWXYZ");
  x = (XX *) s;
  memcpy(x+1, x, sizeof(XX));
  printf("%s\n", s);

  strcpy(s, "ABCDEFG");
  j = x->i;
  x->c1++;
  x->c2++;
  x->i++;
  printf("%s\n", s);

  x->i += (1 << 8);
  x->i += (1 << 17);
  printf("%s\n", s);
  printf("0x%x\n", x->i - j);
}
```

Question 4:

What is the output of this program?

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

main()
{
  int i, j;

  i = 0x12345;
  j = 0xf0f0f;

  printf("0x%x\n", i >> 8);
  printf("0x%x\n", j << 2);
  printf("0x%x\n", i & j);
  printf("0x%x\n", i | j);
  printf("0x%x\n", i ^ j);
  printf("0x%x\n", j | (j << 4));
  printf("0x%x\n", (i << 12) | j);
}
```

Useful Prototypes:

```
void memcpy(void *to, void *from, int bytes);
void strcpy(char *to, char *from);
void strcat(char *to, char *from)
int strlen(char *string);
```

Question 5:

When the procedure **a()** is called, the 48 bytes of memory starting at address 0x100130 are as shown to the right, in hexadecimal. What is the output of **a()** when **x**=0x100130?

Address:	Value:
0x100130	0x10014c
0x100134	0x100130
0x100138	0x100130
0x10013c	0x100150
0x100140	0x100148
0x100144	0x100138
0x100148	0x10013c
0x10014c	0x100138
0x100150	0x100130
0x100154	0x10014c
0x100158	0x100144
0x10015c	0x10014c

```
typedef unsigned long UL;

void a(int *x)
{
  int i, **j;

  for (i = 0; i < 4; i++) {
    printf("i=%d  x+i=0x%lx     x[i]=0x%lx\n",
             i, (UL) (x+i), (UL) x[i]);
  }
  printf("\n");

  j = (int **) x;

  for (i = 0; i < 4; i++) {
    printf("i=%d  j+i=0x%lx    *j[i]=0x%lx\n", i, (UL) (j+i), (UL) *j[i]);
  }
  printf("\n");

  for (i = 0; i < 4; i++) {
    j = (int **) (*x);
    printf("i=%d   j=0x%lx     **j=0x%lx\n", i, (UL) j, (UL) **j);
    x = (int *) j;
  }
  printf("\n");
}
```

Question 6:

When the procedure **a()** is called, the 48 bytes of memory starting at address 0x4d5750 are as shown to the right, in hexadecimal. What is the output of **a()** when **y**=0x4d5750?

Address:	Values in Hexadecimal	Values shown as printable chars:
0x4d5750	0x4d5760	'`' |'W' |'M' |'\0'
0x4d5754	0x4d5764	'd' |'W' |'M' |'\0'
0x4d5758	0x4d5764	'd' |'W' |'M' |'\0'
0x4d575c	0x4d5754	'T' |'W' |'M' |'\0'
0x4d5760	0x4d576c	'l' |'W' |'M' |'\0'
0x4d5764	0x4d577c	'|' |'W' |'M' |'\0'
0x4d5768	0x4d5760	'`' |'W' |'M' |'\0'
0x4d576c	0x4d576c	'l' |'W' |'M' |'\0'
0x4d5770	0x4d5774	't' |'W' |'M' |'\0'
0x4d5774	0x4d5750	'P' |'W' |'M' |'\0'
0x4d5778	0x4d5758	'X' |'W' |'M' |'\0'
0x4d577c	0x4d5770	'p' |'W' |'M' |'\0'

```
typedef unsigned long UL;

void a(char *y)
{
  int i, j;
  char *x;

  x = y;
  for (i = 0; i < 4; i++) {
    memcpy(&j, x, 4);
    printf("i=%d  x=0x%lx  j=0x%lx   x=%s\n", i, (UL) x, (UL) j, x);
    *x = 'a'+i;
    x += 5;
  }

  printf("\n");
  for (i = 0; i < 4; i++) {
    printf("i-%d  y+4*i=%s\n", i, y+4*i);
  }
}
```