

Homework Assignment 2

CSE 399 C++, Spring 2008

Name:

Due: Wednesday, Jan 30th at noon.

Assumptions: For all of these problems you may assume the following

- `sizeof(int) = 4`; `sizeof(short) = 2`; `sizeof(char) = 1`; all pointers require 4 bytes
- The stack starts at address 100 and grows up.
- The heap starts at address 400 and grows up.
- `???` represents an unknown/uninitialized value.

Question 1 (10 points) : Given the following declarations:

```
char c = 'A';
char * p = &c;
char ** p2 = &p;
void * v = &p2;
```

Examine each of the following expressions. If the expression is illegal, write ILLEGAL. If the expression is legal, write its type (i.e. `int **` or `unsigned long` etc):

- `&p2`
- `p2[2]`
- `p + 4`
- `&p2[4]`
- `v[4]`

Question 2 (5 points): There is an old C programmer's joke which goes as follows:

Two strings walk into a bar. The first one says

```
Hi I'd like a beer.A2%asd$ASDlk2;3423Ammm.234ASDfmmlASDFLJ:#@$
```

The second says

```
You'll have to pardon my friend, he's not null terminated.
```


Question 5 (35 points): Consider the following code:

```
void * copy(int * dst, int * src, int count) {
    /* dst is at 124, src is at 128, count is at 132 */
    /* Location 2*/
    while (count) {
        count --;
        dst[count] = src[count];
    }
    /* Location 3 */
    return src + 1; /* be careful ... */
}
...

int a[3]; /* a[0] is at 100 */
int b = 2; /* b is at 112 */
int * x; /* x is at 116 */
int * p = malloc (2 * sizeof (*p)); /* p is at 120 */
a[0] = 9; a[1] = 22; a[2] = 112;
x = &a[1];
/* Location 1 */
x[0] = 33;
x[1] = 99;
x = copy (p, x, b);
/* Location 4 */
x[-1] = 4;
x[0] = 5;
/* Location 5*/
```

Fill in the following table indicating the values of each variable/memory location at each marked program point above. Some boxes are filled in for you.

	Loc 1	Loc 2	Loc 3	Loc 4	Loc 5
100-103	9				
104-107	22				
108-111	112				
112-115	2	2			
116-119					
120-123	400	400			
124-127	???			???	???
128-131	???			???	???
132-135	???			???	???
400-403	???	???			
404-407	???	???			

Question 6 (5 pts): Consider the following code:

```
int a[2];
int b;
int i;
b = 5;

for (i = 0; i <= 2; i++) {
    a[i] = i;
}
printf("b is %d\n", b);
```

When the program is run, the value of `b` mysteriously changes from 5 to 2, even though no assignment is made to `b`. Explain why this occurs:
