CS 31 Homework 4: C Pointers – due March 1 at start of class

Your Name(s)/Lab Section(s):

Question 1

Consider the following declarations and assignments.

```
int *a, b[5], c, *d;

for (c = 0; c < 5; c++)  {
    b[c] = 1+c;
}
d = b;
a = &c;
c = d[3];
```

What are the TYPE and VALUE of each of the following expressions (if the expression is invalid, write “Illegal Expression”, and if it is an address describe what it is the address of):

		TYPE	VALUE
1.	a	------------	-----------
2.	b	------------	-----------
3.	c	------------	-----------
4.	&b[1]	------------	-----------
5.	d	------------	-----------
6.	*d	------------	-----------

Question 2

Answer questions about this C code on the next page.

```
int *fun(int *a, int *b, int s);

void main () {
    int *arr = NULL, x = 6, y = 7, i;

    arr = fun(&x, &y, 5);
    printf("x = %d  y = %d\n", x, y);
    if(arr != NULL) {
        for(i=0; i < 5; i++) {
            printf("arr[%d] = %d\n",
                    i, arr[i]);
        }
    }
    free(arr);
}
/************************************/
int *fun(int *a, int *b, int s) {
    int *tmp, i;

    tmp = malloc(sizeof(int)*s);
    if(tmp != NULL) {
        for(i=0; i < s; i++) {
            tmp[i] = i + *b;
        }
        *a = tmp[2];
        *b = 8;
    }
    // DRAW MEMORY WHEN YOU GET HERE
    return tmp;
}
```

Trace through the C code on the previous page and draw memory contents (heap **and** stack) at the execution point indicated in `func`, and show the output produced by a complete run of the program. (Assume `stdio.h` and `stdlib.h` have been included, and that `malloc` succeeds.)

```
OUTPUT                                                    MEMORY
------                                                    ------
```