

NAME _____

(1) (15 points) Print the values resulting from the following code fragment. **Be careful: at least one of these statements has an unusual effect on a later statement. SHOW WORK FOR PARTIAL CREDIT!!!**

```
int x = 2, y = 3, z[10] = {5, 8, 11, 14, 17, 20, 23, 26, 29, 32}, *ip;
```

```
ip = &z[2];
```

```
y = ip[2];
```

```
printf ("%d, %d",y,*(&z[3]+1)); ANSWER:
```

```
x = *(ip++)+7;
```

```
printf ("%d", *(ip+1)); ANSWER:
```

```
*ip = 0;
```

```
ip = &z[0]+3;
```

```
printf ("%d, %d", *ip, x); ANSWER:
```

(2) (15 points) Consider the struct declaration cont2 with member pointers to two character strings:

```
struct cont2 { char *str1; char *str2;;};
```

Write the code for the function newcont2(), with functional prototype:

```
struct cont2 * newcont2(char * s1, char *s2);
```

Your function should malloc space for a cont2 struct and then needed space for each of the two strings pointed to in the struct, copying into them the valid string arguments s1 and s2 provided by the caller, then return the pointer to the new cont2 struct. If you wish, you can use strdup in K&R to malloc and copy the strings, but if you do this copy the function definition here. If any of your mallocs fail, you must free all the space you have already malloced and return NULL. Note that you are free to use string.h functions such as strlen and strcpy, but you must #include the proper header(s)!

