C Programming

Tutorial 6

1. We created a list of linked nodes using malloc(). When you do free(ptrnode1), where ptrnode1 is the pointer to the first node in the list, do you free the memory for the whole list or only for the first node?

2. When you do structure assignment, is there any thing that potentially worries you? (for example, a pointer element is involved)

3. I used the statement (int *p = malloc(5);) in two programs. It worked for one but not for the other. What could be the problem?

4. Do you see any problem with this statement: char *p = malloc(strlen(s));?

5. Any thing wrong with the following statement, trying to dynamically allocate an array?

```
char *char_array = malloc(num_chars);
int *int_array = malloc(num_ints);
```

6. Is the free'ed memory returned to the OS?

7. See the following code. What is wrong with it? How would you fix the problem?

```
char *func(int n)
{
        char a[10];
        … /* do something about a */
        return a;
}
```

8. See the following statement. Gives the possible returns that the statement could give.

int *tmp = realloc(ptr, 20 * sizeof(int));

Can I do int *tmp = realloc(ptr, 10 * sizeof(int)); int *tmp = realloc(ptr, 0);

9. Do you really want to free the memory just before finishing the execution of the program?

10. Discuss how to create dynamic arrays, one-dimensional, two-dimensional, three-dimensional, …. Are they really arrays?