

ECE160B - Programming Languages for Electrical Engineers
Spring 2018
03/01/18
Quiz #3

Name: _____

Score: ____/10

1. Reading Pointer Code - Fill out the table to the right of the code with the value of the variable specified. Each row is 0.5 points. Here is a conversion table that may be useful.

0 °C	32.0 °F		40 °C	104.0 °F		80 °C	176.0 °F
10 °C	50.0 °F		50 °C	122.0 °F		90 °C	194.0 °F
20 °C	68.0 °F		60 °C	140.0 °F		100 °C	212.0 °F
30 °C	86.0 °F		70 °C	158.0 °F			

Code	ptr_F	ptr_C	*ptr_F	*ptr_C
<pre>#include <stdio.h> int main() { double temp_in_F = 32.0; // temperature in Fahrenheit double temp_in_C = -100.0; // temperature in Celsius double *ptr_F; double *ptr_C;</pre>				
<pre> ptr_F = &temp_in_F; // address is 0xffffcbd8 ptr_C = &temp_in_C; // address is 0xffffcbd0</pre>				
<pre> *ptr_C = *ptr_F;</pre>				
<pre> *ptr_C = (5.0/9.0) * (*ptr_F - 32.0);</pre>				
<pre> *ptr_F = 212.0;</pre>				
<pre> ptr_C = ptr_F;</pre>				
<pre> temp_in_C = 50.0;</pre>				

Code (continued)	array_ptr_C	array_ptr_F	*array_ptr_C	*array_ptr_F
<pre> double temp_array_in_C[] = { 0.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0, 70.0, 80.0, 90.0, 100.0 }; double temp_array_in_F[11]; for (int i = 0; i < 11; i++) { temp_array_in_F[i] = 9.0/5.0 * temp_array_in_C[i] + 32.0; } double *array_ptr_C; double *array_ptr_F; </pre>				
<pre> array_ptr_C = temp_array_in_C; array_ptr_F = &temp_array_in_F[5]; </pre>				
<pre> *array_ptr_C = 110.0; </pre>				
<pre> *array_ptr_F = 212.0; </pre>				
<pre> array_ptr_C = array_ptr_F; </pre>				
<pre> // Bonus (+0.5 points) *(array_ptr_F++) = 77.7; return 0; } </pre>				

2. Writing Code Using Pointers

Write a pointer version of `strncmp()`. The following is the documentation from K&R pg. 249:

`int strncmp(cs, ct, n)` - compare at most `n` characters of string `cs` to string `ct`; return `<0` if `cs<ct`, `0` if `cs==ct`, or `>0` if `cs>ct`.

This is the function prototype:

```
int strncmp(char *cs, char *ct, int n);
```

For example:

`strncmp("ABCF", "ABCDEF", 3)` returns 0

`strncmp("ABAC", "ABCDEF", 3)` returns -2

`strncmp("ABDC", "ABCDEF", 3)` **returns 1**