

EE 1910

Dr. Johnson

Homework 9

1 – Given the following memory map – evaluate each item

20pts

```
int foo;  
int boo;  
int zoo;  
int* a_ptr;  
int* b_ptr;
```

variable name	value	address
foo	0x1234	0x1000
boo	0x8000	0x2000
zoo	0x2324	0x3000
a_ptr	0x2000	0x7000
b_ptr	0x3456	0x8000

&boo addr of -> 2000

*b_ptr value pointed to by b_ptr (value at 0x3456)

a_ptr value of a_ptr

&b_ptr addr of b_ptr

boo + b_ptr + *a_ptr + &boo

$$8000 + 3456 + 8000 + 2000$$

HEX

0x2000
?????
0x2000
0x8000
0x15456

2 – Fill in the memory map at the end of the following code

40pts

*** note: this code will not compile – for illustrative purposes only ***

```
int foo, boo;  
float zoo, soo;  
int* a_ptr, b_ptr;  
float* c_ptr, d_ptr;
```

```
a_ptr = &boo;  
c_ptr = a_ptr + 0x2000;  
*c_ptr = 3.5 * foo;  
*a_ptr = 7;  
zoo = soo + boo;  
d_ptr = a_ptr + c_ptr;  
*d_ptr = 0x1000;  
*b_ptr = boo + *a_ptr;
```

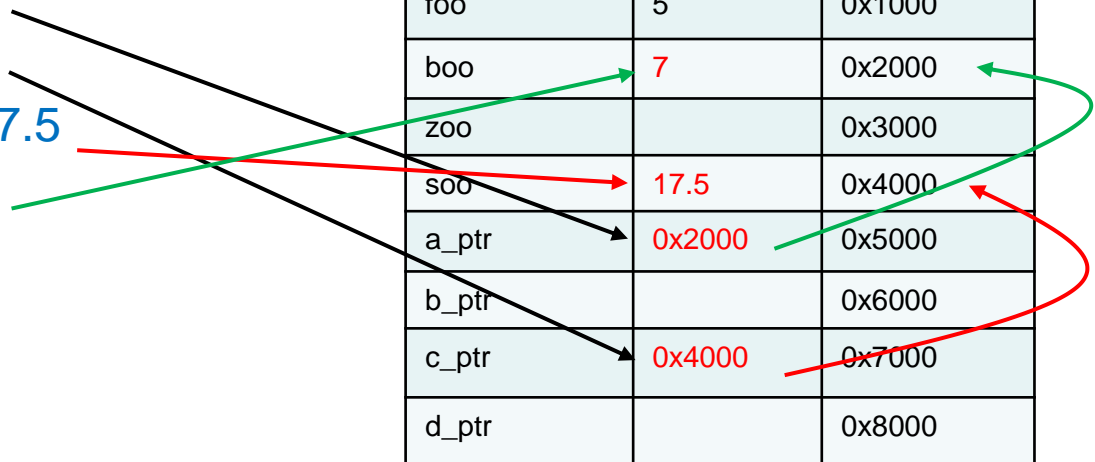
0x2000

0x4000

soo = 17.5

boo = 7

variable name	value	address
foo	5	0x1000
boo	7	0x2000
zoo		0x3000
soo	17.5	0x4000
a_ptr	0x2000	0x5000
b_ptr		0x6000
c_ptr	0x4000	0x7000
d_ptr		0x8000



2 – Fill in the memory map at the end of the following code

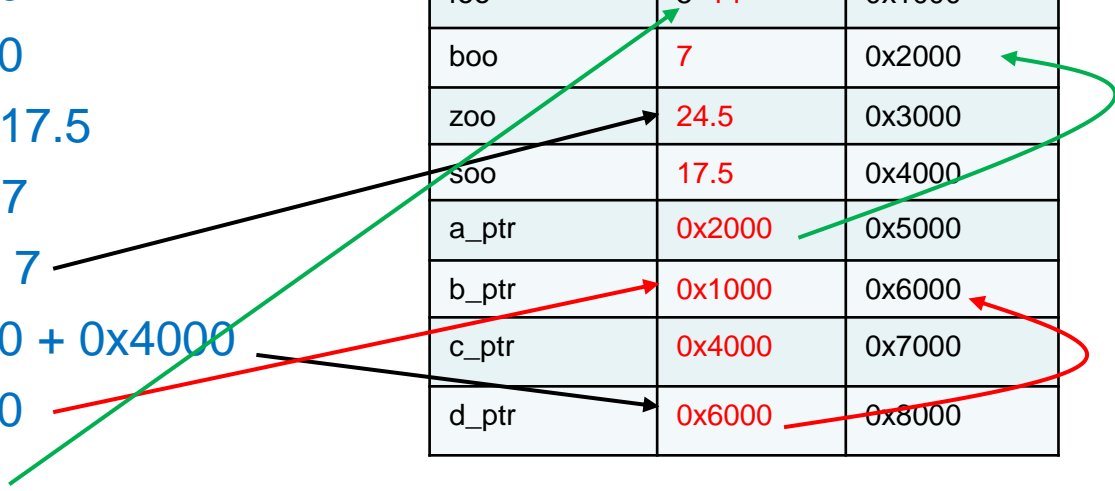
40pts

*** note: this code will not compile – for illustrative purposes only ***

```
int foo, boo;
float zoo, soo;
int* a_ptr, b_ptr;
float* c_ptr, d_ptr;
```

```
a_ptr = &boo;           0x2000
c_ptr = a_ptr + 0x2000; 0x4000
*c_ptr = 3.5 * foo;     soo = 17.5
*a_ptr = 7;             boo = 7
zoo = soo + boo;       17.5 + 7
d_ptr = a_ptr + c_ptr; 0x2000 + 0x4000
*d_ptr = 0x1000;       0x1000
*b_ptr = boo + *a_ptr; 7 + 7
```

variable name	value	address
foo	5 14	0x1000
boo	7	0x2000
zoo	24.5	0x3000
soo	17.5	0x4000
a_ptr	0x2000	0x5000
b_ptr	0x1000	0x6000
c_ptr	0x4000	0x7000
d_ptr	0x6000	0x8000



3 – Provide the final values after executing the following code snippet 40pts

```

#include <stdio.h>

int fun1(int* a, int* b);
float fun2(float a, float* b);

int main(void){

    int a = 3;
    int b = 2;
    float c = 3.5;
    float d = 4.5;
    float e = 12;

    a = fun1(&a,&b);
    e = fun2(c,&d);

    return 0;
}

int fun1(int* foo, int* boo ){
    int zoo;
    if (*boo > 0){
        zoo = 2**foo;
    }
    else{
        zoo = 3**foo;
    }
    *boo = zoo + *foo;
    *foo = 12;
    return zoo;
}

float fun2(float zoo, float* soo){
    zoo = 34;
    *soo *= 3*zoo;
    return *soo;
}

```

a 3 -> 12 -> 6
 b 2 -> 9
 c 3.5
 d 3.5 -> 459.0
 e 12 -> 459.0

a	6
b	9
c	3.5
d	459.0
e	459.0