1. a. 8
b. 48
c. 12 because its a pointer pointer
d. 4
e. they point at the same address
f. (on last page of pdf)

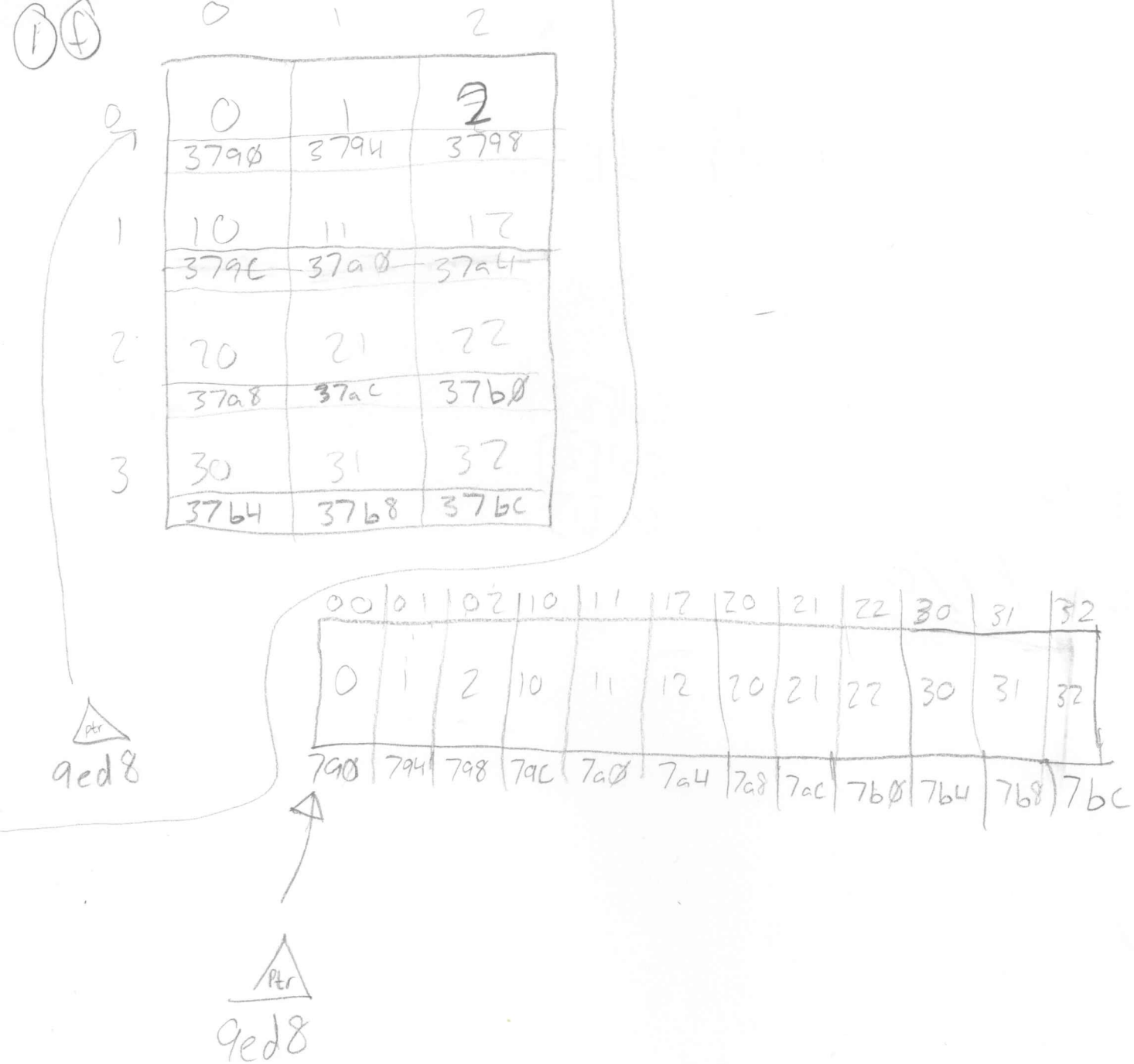
3. a. all correct guesses

```
twod + 3 is: 0x7fff5871fbc4
*(*(twod + 1)) is: 10
*twod + 1 is: 0x7fff5871fba4
*twod[2] is: 20
*(twod + 2) + 2 is: 0x7fff5871fbc0
twod[1] is: 0x7fff5871fbac
twod[1][2] is: 12

ptr 0x7fff5871fba0
twod [1] 0x7fff5871fbac
ptr[1] 1
ptr + 1 0x7fff5871fba4
*(ptr + 1) 1
twod + 1 0x7fff5871fbac
*twod + 1 0x7fff5871fba4
ptr[8] 22
```

4. no, because that is targeting a pointer that does not exist. if it was adding too much that moved out of the array then it would but this is an unknown location that we do not own

11. int (*twod)[3] as a parameter is equivalent to passing in int twod[][3]. whereas passing in int *(twod[3]) is passing in the * array that is stored in row 3.


0 1 2
0
1
2
3
3790 3794 3798
10 11 12
379c 37a0 37a4
20 21 22
37a8 37ac 37b0
30 31 32
3764 3768 37bc
00 01 02 10 11 12 20 21 22 30 31 32
0 1 2 10 11 12 20 21 22 30 31 32
ptr
9ed8
790 794 798 79c 7a0 7a4 7a8 7ac 7b0 7b4 7b8 7bc
Ptr
9ed8

Answer /complete the following

a) What is the size of ptr?
b) What is the size of twod?
c) What is the size of twod[0] and why?
d) What is the size of twod[0][0]?
e) What can you say about twod and twod[0] as it relates to the name of the array?
f) Draw a memory map that shows the memory locations of each element of the array and of ptr.

2) Using the provided address from #1 as the base address of the 2D array and the location of ptr, based on the code below, create an educated guess that clearly outlines what you believe will happen as each line is executed. In your explanation clearly explain what is happening, don't just give memory addresses or values. If you only provide memory addresses or values you will receive 0 points for this problem. Your guesses will be clearly labeled in the PDF file. You must provide the line of code and then the explanation. You must also provide per each line of code what the type is. Within each printf statements is a ?. You must specify if the ? would be a **d** for an int or a **p** for a pointer.

printf("twod + 3 is: %?\n", twod + 3); Add 3 ints to address twod - int* - 3764
printf("*(*(twod + 1)) is: %?\n", *(*(twod + 1))); Add 1 int to add twod - int - 10
printf("*twod + 1 is: %?\n", *twod+1); Add 1 int to the * at [0] - int* - 3794
printf("*twod[2] is: %?\n", *twod[2]); Value at row 2 - int - 20
printf("*(twod + 2) + 2 is: %?\n", *(twod + 2) + 2); Add 2 ints to rows, then 2 ints to cols - int* - 3760
printf("twod[1] is: %?\n", twod[1]); Address at row 1 array start - int* - 379c
printf("twod[1][2] is: %?\n", twod[1][2]); Value at row 1, col 2 - int - 12

printf("ptr %?\n", ptr); int* - 9ed8 - ptr address
printf("twod [1] %?\n", twod [1]); Address of row 1 array start - int* - 379c
printf("ptr[1] %?\n", ptr[1]); move ptr through row 0 to col 1 - move 1 int - int - 1
printf("ptr + 1 %?\n", ptr + 1); add 1 int to ptr - int* - 3794
printf("*(ptr + 1) %?\n", *(ptr + 1)); add 1 int to ptr, then get value - int - 1
printf("twod + 1 %?\n", twod+1); add 1 int to twod - address of twod[1] - int* - 379c
printf("*twod + 1 %?\n", *twod + 1); Add 1 int to the * at twod[0] - int* - 3794
printf("ptr[8] %?\n", ptr[8]); Add 8 ints to the address of ptr, get value - int - 22