

Section: 4 Name: Jashan Rai

Problems 1-10 refer to the following statements:

```
/*      0  1  2  3  4  5  array positions */
int x[] = {5, 6, 4, -8, 3, 7};
int *ptr = &x[0];
```

What is the value of the following expressions? For each problem, restart with the values as above.

	<u>Work Space</u>	<u>Your Answer</u>	<u>Computer</u>
1. *ptr	_____	1. <u>5</u>	1. <u>5</u>
2. *ptr + 3	_____	2. <u>8</u>	2. <u>8</u>
3. *(ptr+3)	_____	3. <u>-8</u>	3. <u>-8</u>
4. *ptr + *(ptr + 5)	_____	4. <u>12</u>	4. <u>12</u>
5. *(ptr + 2) - 1	_____	5. <u>3</u>	5. <u>3</u>
6. x[3] - *ptr	_____	6. <u>-13</u>	6. <u>-13</u>
7. *ptr + x[5] + *(ptr + 1) + x[2]	_____	7. <u>22</u>	7. <u>22</u>
8. *x	_____	8. <u>5</u>	8. <u>5</u>
9. *x + *ptr	_____	9. <u>10</u>	9. <u>10</u>
10. x[2] - *ptr + 3	_____	10. <u>2</u>	10. <u>2</u>

➔ more on next page

Problems 11-16 refer to the following declarations and function:

```
int partial_sum (int x[], int npts); /* function prototype */

/* Array & variables as initialized in main, abridged */

int main (void)
/*      0  1  2  3  4  5  6  7  array positions */
int a[ ] = {-6, 3, 4, 1, 8, 20, 16, 7};
int *ptr = &a[2];
partial_sum( see below);

/*-----*/
/* This function will add up a fragment of the array */
int partial_sum (int x[], int npts) {

    int k, sum = 0;

    /* Compute partial sum. */
    for (k = 0; k < npts; k++)
        sum += x[k];

    return sum;
}
/*-----*/  /* workspace below */
```

	<u>You</u>	<u>Computer</u>
11. What is the value of the reference partial_sum(ptr, 2) _____	11. <u>5</u>	11. <u>5</u>
12. What is the value of the reference partial_sum(ptr+1, 3) _____	12. <u>29</u>	12. <u>29</u>
13. What is the value of the reference partial_sum(a, 8) _____	13. <u>53</u>	13. <u>53</u>
14. What is the value of the reference partial_sum(a, 4) _____	14. <u>2</u>	14. <u>2</u>
15. What is the value of the reference partial_sum(ptr, a[1]) _____	15. <u>1</u>	15. <u>13</u>
16. What is the value of the reference partial_sum(&a[3], 2) _____	16. <u>9</u>	16. <u>9</u>


→ more on next page

Line 1. `int y[] = {2, 13, 5, 17, 8, 6, 15}, *ptr = &y[3];` `// could also do: int *ptr = y+3;`

Line 2. `*ptr = *ptr + 4;`

Line 3. `*(ptr+2) = y[0];`

Questions:

17. What is the value of `*ptr`, after initialization, after line 1; 17. 17
18. What is the value of `*ptr` after the execution of line 2? 18. 21
19. What is the value of `*(ptr+2)` after the execution of line 3? 19. 2
20. What are the values in the whole array **after** all three lines of code have been executed?
-  2 13 5 21 8 2 15

FILE you need

The file you need for lab6 to fill in the “computer” part is: `lab6.c`

- To get it from the Coding computers:
 - First move to your class folder by typing: `cd csc60`
 - The long command below will create a `lab6` directory and put one file in it.
 - Type: `cp /home/college/bielr/files_csc60/lab6.c .` (Don’t miss the “space dot” after the `c`)
 - Next the file permissions need to be changed by typing: `chmod 644 lab6.c`
 - Now you can move into the `lab6` directory
- You can also download it from Canvas.

Compile, run it, fill in the *Computer Section* of the worksheet.

No points off for wrong guesses.

The point is to learn from both the correct answers and the wrong ones.

Turn in this worksheet for credit:

- If you worked on a paper copy, scan the papers and then submit the PDFs to Canvas.
- If you worked electronically in Adobe, be sure to Save the File (upper left corner), and then submit it to Canvas.

Total Score = **20 points** If turned in with the sheet filled in. Not grading for right or wrong.