

ASSIGNMENT: C5 Bonus

FILE: C5_ACT_katherto.pdf

DATE: 17 February 2016

BY: Kathryn Atherton

katherto

Joshua Hahn

Hahn28

Hannah Mackin Schenck

hmackins

SECTION: 03, 1:30-3:30

TEAM: 45

ELECTRONIC SIGNATURE:

Kathryn Atherton

Joshua Hahn

Hannah Mackin Schenck

The electronic signatures above indicate that the document submitted for evaluation is the combined effort of all team members and that each member of the team was an equal participant in its creation. In addition, each member of the team has a general understanding of all aspects of the document.

TASK 1

PART A:

1. The value of c is 'S'. The value of d is 'T'. The value of e is 'S'. The value of &c is 6940.
2. The value of p1 is 9772. The value of p2 is 6940. The value of p3 is 2224.
3. The value of *p1 is 'T'. The value of *p2 is 'S'. The value of *p3 is 'S'.

Code: C5_ACT_Task1a_katherto.c

4. The values obtained by hand are the same, except for the pointer values, because the addresses on the computer are different from the given values.
5. The values *p1, *p2, and *p3 are called "dereferenced pointer values" and the values p1, p2, and p3 are pointer values.
6. & returns the address of a variable. * returns the variable that an address points to.

PART B:

1. Temp = 'A'
2. Temp = 'C'
3. Temp = 'B'
4. Temp = 'A'
5. Ptr = 4435
6. Temp = 'B'
7. Temp = 'C'
8. Ptr = 4434
9. Temp = 'B'

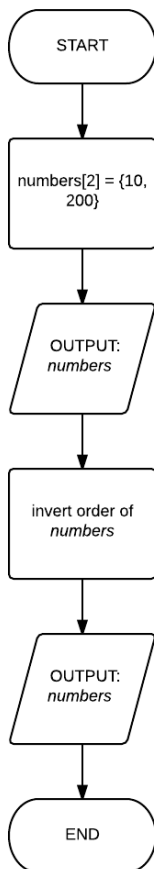
10. Temp = 'C'
11. Temp = 'C'
12. Temp = 'C'
13. Ptr = 4435

Code: C5_Task1b_katherto.c

TASK 2:

PART A:

Flowchart:



PART B:

Code: C5_ACT_Task2b_katherto.c

PART C:

Code: C5_ACT_Task2c_katherto.c

1. Yes, the original array in main was changed.
 2. Passing the pointer allows the program to work on the memory it is stored in, thus changing the original variable.
-

TASK 3:

Code: C5_ACT_Task3_katherto.c

1. It is possible to use pointer arithmetic with an array because an array variable is a pointer.
-

TASK 4:

PART A:

Flowchart: Next page

PART B:

Code: C5_ACT_Task4_katherto.c

