ANSWER KEY

1. **A**	16. **B**	31. **A**	46. **C**
2. **C**	17. **A**	32. **C**	47. **B**
3. **A**	18. **D**	33. **A**	48. **C**
4. **D**	19. **A**	34. **C**	49. **A**
5. **B**	20. **C**	35. **D**	50. **D**
6. **B**	21. **A**	36. **C**	51. **C**
7. **D**	22. **A**	37. **A**	52. **D**
8. **C**	23. **D**	38. **D**	53. **D**
9. **C**	24. **B**	39. **B**	54. **A**
10. **B**	25. **C**	40. **C**	55. **C**
11. **D**	26. **A**	41. **B**	56. **D**
12. **C**	27. **D**	42. **B**	57. **B**
13. **C**	28. **A**	43. **A**	58. **A**
14. **B**	29. **C**	44. **C**	
15. **B**	30. **B**	45. **D**	

ANSWERS EXPLAINED

1. **(A)** When tracing an algorithm, using a trace table makes keeping track of your variables easier. Notice the display is *c*, then *a*, and then *b*. Don't assume the order is always *a*, *b*, *c*. Always evaluate the right-hand side of the equation first, and then set the value to the variable on the left.

a	*b*	*c*	Output
~~13~~	17	2	2 14 17
14			

2. **(C)** When tracing an algorithm, using a trace table makes keeping track of your variables easier. Although *a* was initially set to 13, it is overwritten in the second line, setting *a* equal to 17. In the third line, 1 is added to *a*, setting it equal to 18.

a	Output
~~13~~	18
~~17~~	
18	