Fixed on November 8, 2016

ECE 30862 Fall 2014 Final Exam Answer Sheet
Both sides of the sheet must be filled in

All answers should be on the **front and back** of this sheet. Both this answer sheet and your test must be signed and turned in. All questions are worth 1.3 points.

I promise that I have neither given nor received disallowed aid on this test.

Name (Printed): Name (Signed):

1.	b	**20.**	dil
2.	B D	**21.**	B final D final
3.	bll	**22.**	b
4.	dil	**23.**	a
5.	dil	**24.**	c
6.	D	**25.**	c
7.	B	**26.**	E
8.	bll	**27.**	Di
9.	bil	**28.**	Di
10.	bil	**29.**	Di
11.	B	**30.**	Di
12.	B	**31.**	Di
13.	D	**32.**	Bi
14.	B	**33.**	Bi
15.	B	**34.**	Di
16.	dil	**35.**	Bi
17.	dil	**36.**	Di
18.	B	**37.**	Bi
19.	B	**38.**	Bi

39. E

40. Di

41. Di

42. Bi

43. Bi

44. Bi

45. OK

46. E

47. OK

48. OK

49. E

50. E

51. E

52. E

53. 0 0 4

54. B

55. B D

56. $\sim$ B $\sim$ C *or* B D $\sim$ B $\sim$ C

57. $\sim$ B $\sim$ C

58. $\sim$ B $\sim$ C

59. LINE B

60. LINE A

61. I I 20 10

62. 10 20

63. 2 1

64. 2 1

65. 1 2

66. 1 2

67. b

68. 1 3

69. 1 2

70. -1

71. c

72. a

73. -1 2

74. 1 2

75. c

B* b = new B( ); // Q54 B
Call the B constructor which prints **B**. The B constructor constructs a C object, but the C constructor doesn't print anything.

D* di = new D(1); // Q55 B D
Call the D constructor, which immediately calls the base class zero arg B constructor as part of its execution of the initializer list. This prints **B**. The D constructor body then executes and prints **D**.

D dv = D(1); // NOT part of a question
It will print **B D** for the same reason as given for Q55

delete b; // Q56 ~B ~C
When the B destructor is called it prints **~B** and then deletes the c object, which causes the C destructor to be called, which prints **~C**.

delete di; // Q57 ~B ~ C
This will call the D destructor to be called, since di points to a D object. A default D destructor is called, which doesn't print anything. [As a side note, I don't generally like relying on default destructors.) Since D inherits from B, the last thing the D destructor does before exiting is call ~B, the B destructor. This then prints out **~B ~C** for the reasons given in Q56.

} // Q58
dv, which is a D object, is popped off the stack. When this happens the D object destructor is called and **~B ~C** is printed for the reasons given in Q56.