WYSE – Academic Challenge
Computer Science Test (Sectional) – 2012
Solution Set

1. **Correct Answer: E.**

a – SQL is used to define tables
b – SQL is used to created relationships between tables
c – SQL is used to insert rows into tables
d – SQL is used to update existing data with a database
e – SQL is used for all of the above operations

2. Correct Answer: D.

All three encoding forms (UTF-8, UTF-16, & UTF-32) need at most 4 bytes (or 32-bits) of data for each character.

3. Correct Answer: B.

The bank routing number, account number, and the amount are written with magnetic ink on the bottom of deposit slips and checks. The magnetic ink allows the numbers to be read more reliably than traditional optical character recognition systems. With magnetic ink, the numbers can still be read even if the numbers have been overwritten with handwriting or cancellations stamps.

4. Correct Answer: B.

CMOS is the acronym for Complementary Metal Oxide Semiconductor.

5. Correct Answer: D.

A normal 8 bit binary number has values of 0 through 255. The original range is of 0 through 255 is then shifted to -127 through 128 by subtracting the bias of 127.

6. Correct Answer: E.

10100101 $= -1*2^7 + 0*2^6 + 1*2^5 + 0*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 1*2^0$
$= -128 + 32 + 4 + 1$
$= -91$

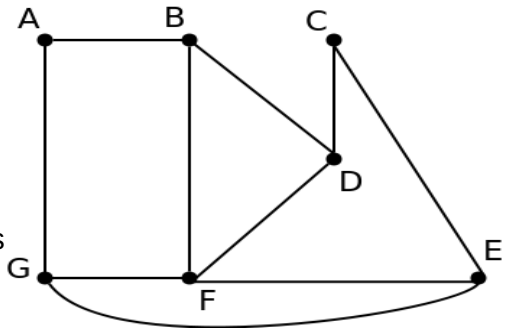
7. Correct Answer: D.

(A AND B)' = A' OR B' This is DeMorgan's Law.

8. Correct Answer: B.

(A XOR TRUE) OR (A(TRUE)C')	Replace TRUE for B
A' OR (A(TRUE)C')	A XOR TRUE is only TRUE when A is FALSE
A' OR (AC')	AC' anded with TRUE is still AC'

9. Correct Answer: E.



A depth first search moves forward until it cannot reach a node that has not be visited yet. Then it backtracks until it reaches a node that is connected to other nodes that have not been visited.

A → B → D → C → E → F → G

In this case, no backtracking takes place. Note that several correct answers exist for this problem, but only one of the correct answers is provided. For example, AGFECDB is also a correct dept first search. Another popular search for a graph is a breadth first search where all nodes that can be reached from the start node are searched first and then this process repeats until the graph is totally searched. ABGDFEC is an example of a breadth first search.

10. Correct Answer: E.

INSTRUCTION	STACK
PUSH(A);	A
PUSH(B);	AB
PUSH(C);	ABC
POP();	AB
PUSH(D);	ABD
PUSH(E);	ABDE
PUSH(F);	ABDEF
POP();	ABDE

11. Correct Answer: B.

The % sign is the Modulus operator

12. Correct Answer: C.

The `for` statement will be the next statement after the continue. Note that the `continue` causes the loop to stop where it is at and return to the top of the loop. A `break` statement would cause the loop to terminate.

13. Correct Answer: D.

A trace of the code follows.

At beginning of loop		if (i%3 == 0)	if i%3 != 0	at end of loop
i	**j**	**j--**	**j += 2;**	**j**
0	0	y	n	-1
1	-1	n	y	1
2	1	n	y	3
3	3	y	n	2
4	2	n	y	4
5	4	n	y	6
6	6	y	n	5
7	5	n	y	7
8	7	n	y	9
9	9	y	n	8
10	8	n/a	n/a	n/a

The code will output 10 8.

14. Correct Answer: D.

Leaves are nodes without any children. The tree has four leaves.

15. Correct Answer: C.

A bubble sort will perform 99 comparisons on the first pass. The second pass will require 98, etc.
99 + 98 + 97 + … + 2 + 1 = (99*100)/2 (The simplification uses Gauss's formula).

16. Correct Answer: A.

The code segment has a nested loop.

17. Correct Answer: C.

A trace of the code is below:

i	j	inner loop output (j)	outer loop output	
0	0	0	endl	inner loop is executed once (j<=i)
1	0 1	01	endl	inner loop is executed twice
2	0 1 2	012	endl	inner loop is executed three times
3	0 1 2 3	0123	endl	inner loop is executed four times

18. Correct Answer: D.

A trace of the code is below:

i	j	inner loop output (i)	outer loop output	
0	0	0	endl	inner loop is executed once (j<=i)
1	0	11	endl	inner loop is executed twice
	1			
2	0	222	endl	inner loop is executed three times
	1			
	2			
3	0	3333	endl	inner loop is executed four times
	1			
	2			
	3			

19. Correct Answer: C.

Trace of code follows.

a	b	c	CODE
undef	undef	undef	
			`int a = 5, b = 20, c=0;`
5	20	0	
			`cout << a << '\b' << b << ' ' << c << endl;`
5	**20**	**0**	
			`c = (a < b) ? a: b;`
5	**20**	**5**	
			`cout << a << ' ' << b << ' ' << c << endl;`

The '\b' will cause the 5 that was output from variable a to be overwritten. The answers that are bold represent the output.

20. Correct Answer: A.

See above code trace.

21. Correct Answer: E.

The two lines are function prototypes. Functions are often also called procedures, but this is the prototype for the function. Placing the prototype in this location allows the following code to use the function even though the full definition does not come until later in the program code. The programmer can then keep the main function in the program as the first function seen in the program and let the other user defined functions follow.

22. Correct Answer: C.

The variable a is a global variable and will be updated in all blocks of code unless a variable of the same name is defined within that block. Within this code, main and `my_func2` will both update the global variable. `my_func1` will not update the global variable as the function has a variable of the same name defined within it. Therefore, the correct answer is C—`main` and `my_func2`.

23. Correct Answer: A.

User inputs 2

Global a at beginning of while	Global a after `my_func1`	Global a after `my_func2`
2	4	8
8	64	128

The code will output 128.

24. Correct Answer: A.

The variable a is a global variable that may be accessed in any of the code that follows it. In general, it is common practice to avoid the use of global variables for anything except constants that cannot change. The use of global variables can make debugging larger programs difficult, as it becomes complicated to identify where the global variable may be changed in the larger programs.

25. Correct Answer: C.

The memory for a1 and a2 is dynamically allocated and will be allocated at run time.

26. Correct Answer: B.

25 25 25 will be on the second line of output. See the code trace below.

a1	***a1**	**a2**	***a2**	**a3**	**CODE**
undef	undef	undef	undef	undef	
					`int* a1 = new int;`
a1 mem loc	undef	undef	undef	undef	
					`int* a2 = new int;`
a1 mem loc	undef	a2 mem loc	undef	undef	
					`int a3 = 20;`
a1 mem loc	undef	a2 mem loc	undef	20	
					`*a1 = a3;`
a1 mem loc	20	a2 mem loc	undef	20	
					`*a2 = 25;`
a1 mem loc	20	a2 mem loc	25	20	
					`cout << *a1 << " " << *a2 << " " << a3 << endl;`
a1 mem loc	20	a2 mem loc	25	20	
					`*a1 = 5;`
a1 mem loc	**5**	a2 mem loc	25	20	
					`a1 = a2;`
a2 mem loc	25	a2 mem loc	25	20	
					`a3 += 5;`
a2 mem loc	**25**	a2 mem loc	**25**	**25**	
					`cout << *a1 << " " << *a2 << " " << a3 << endl;`
a2 mem loc	25	a2 mem loc	25	25	
					`a3 += *a2;`
a2 mem loc	25	a2 mem loc	25	50	

					`*a2 = a3;`
a2 mem loc	25	a2 mem loc	50	50	
					`delete a1;`
undef	undef	**undef**	**undef**	50	
					`cout << *a2 << endl;`

27. Correct Answer: E.

The third line of output will display an unknown value. 5 remains in the original memory location allocated to a1. However, this particular memory location cannot be used by the program once the pointer a1 is changed to a2. This results in a memory leak and can be a particularly bad problem with structures such as linked lists, where an entire lengthy list containing significant amounts of memory cannot be used or freed until the program ends. So, once a1 is deleted, a2 is also deleted as both a1 and a2 point to the same memory location. The pointer variable still exist, but the data the memory locations have pointed to has been released back to the computer for other use. The values in the previous locations that the variables pointed to will vary depending upon the compiler, operating system and other programs the system might be running. Because of this, there is no certainty of the value that will be displayed.

28. Correct Answer: D.

All of the methods with the exception of the overloaded ++ operator have their code in the class header file making them inline with the object declaration.

29. Correct Answer: B.

Polymorphism allows different types of objects to use the same function or operator in a different way that is appropriate for that given object. Encapsulation involves encapsulating the data and providing an interface for (get and set methods) the user to access these objects, this prevents the user from altering the objects in a way that is not appropriate for that class (ie. making the age of someone negative). Abstraction allows users of a class to use the class without having to become bogged down in the implementation of the various different aspects of the class. Inheritance allows classes to be built on top of the foundation of other classes and inherit the properties of the base class.

30. Correct Answer: A.

This is a post increment operator. The age is incremented by a value of 1, but the person object returned contains the age before the increment, hence the increment is applied after (post) actually using the value of the person.