2019-2020 | **American Computer Science League** | **ACSL Finals**

Senior Shorts Solutions

1. Boolean Algebra

$$\overline{A + \overline{B}\,C} + \overline{B + \overline{A\,C}} + \overline{\overline{C} + A\,B}$$

$$= \overline{A}\,\overline{\overline{B}\,C} + \overline{B}\,\overline{\overline{A\,C}} + \overline{\overline{C}}\,\overline{A\,B}$$

$$= \overline{A}\,(\overline{\overline{B}} + \overline{C}) + \overline{B}\,A\,C + \overline{\overline{C}}\,(\overline{A} + \overline{B})$$

$$= \overline{A}\,B + \overline{A}\,\overline{C} + \overline{B}\,A\,C + \overline{A}\,C + \overline{B}\,C$$

$$= \overline{A}\,B + \overline{A}\,(\overline{C} + C) + A\,\overline{B}\,C + \overline{B}\,C$$

$$= \overline{A}\,B + \overline{A} + \overline{B}\,C\,(A + 1)$$

$$= \overline{A}\,B + \overline{A} + \overline{B}\,C$$

$$= \overline{A}\,(B + 1) + \overline{B}\,C$$

$$= \overline{A} + \overline{B}\,C$$

There is one OR operator.

1. 1 (B)

2. Boolean Algebra

$A \,\$\, B = (A \oplus \overline{B})(\overline{A} \oplus B)$ simplified is $AB + \overline{A}\,\overline{B}$.

Therefore, $\overline{A} + A \cdot B \,\$\, C$
$= \overline{A} + A(BC + \overline{B}\,\overline{C})$
$= \overline{A} + ABC + A\,\overline{B}\,\overline{C}$.

$\overline{A}$ is TRUE for the ordered triples (0,*,*) and the other two ordered triples are (1,1,1) and (1,0,0) so there are a total of 6.

2. 6 (C)

3. Bit-String Flicking

Change all to binary:

$AB9_{16} = 101010111001$, $CE9_{16} = 110011101001$,
$915_{16} = 100100010101$

((NOT (RCIRC-7 $AB9_{16}$)) XOR (RSHIFT-3 ($CE9_{16}$ AND 915_{16})))
= ((NOT (RCIRC-7 101010111001)) XOR
(RSHIFT-3 (110011101001 AND 100100010101)))
= ((NOT 011100110101) XOR (RSHIFT-3 100000000001))
= 1000 1100 1010 XOR 0001 0000 0000
= 1001 1100 1010
= 9 C A

3. 9CA (B)

4. Bit-String Flicking Let X = abcde and NOT X = ABCDE ((RCIRC-2 (X AND 11011)) OR (RCIRC-2 X) XOR 01110)= (NOT (LSHIFT-4 01011)) ((RCIRC-2 (abcde AND 11011)) OR (RCIRC-2 abcde) XOR 01110) = (RCIRC-2 ab0de) OR (deabc XOR 01110) because XOR has a higher priority than OR does. LHS: deab0 OR dEABc RHS: (NOT (LSHIFT-4 01011)) = NOT 10000 = 01111. Therefore, d + d = 0 so d = 0, e + E = 1 which is always TRUE, a + A = 1 which is always TRUE, b + B = which is always TRUE, and 0 + c = 1 so c = 1. which gives the bit string solution **10* so there are 8 solutions.	4. 8 (D)
5. Recursive Functions	5. 19 (A)

$f(14,20) = f(14+1, 20-2) + f(14,20) + 1$ $= f(15,18) + f(20,14) + 1 = 12 + 6 + 1 = 19$ $f(15,18) = f(15+1, 18-2) + f(18,15) + 1$ $= f(16,16) + f(18,15) + 1 = 8 + 3 + 1 = 12$ $f(20,14) = 20 - 14 = 6$ $f(16,16) = f\left(f\left(\frac{16}{2}, 16\right), \frac{16}{2}\right) - 3 = f(f(8,16), 8) - 3$ $= f(19,8) - 3 = 11 - 3 = 8$ $f(18,15) = 18 - 15 = 3$ $f(8,16) = f(8+1, 16-2) + f(16,8) + 1$ $= f(9,14) + f(16,8) + 1 = 10 + 8 + 1 = 19$ $f(9,14) = f(9+1, 14-2) + f(14,9) + 1$ $= f(10,12) + f(14,9) + 1 = 4 + 5 + 1 = 10$ $f(16,8) = 16 - 8 = 8$ $f(14,9) = 14 - 9 = 5$ $f(10,12) = f(10+1, 12-2) + f(12,10) + 1$ $= f(11,10) + f(12,10) + 1 = 1 + 2 + 1 = 4$ $f(11,10) = 11 - 10 = 1$ $f(12,10) = 12 - 10 = 2$ $f(19,8) = 19 - 8 = 11$	
6. Recursive Functions The pattern is that there is 1 trunk of length 128; there are 2 branches of length 64; there are 4 branches of length 32; there are 8 branches of length 16; there are 16 branches of length 8; there are 32 branches of length 4; there are 64 branches of length 2; and there are 128 branches of length 1. Each product is 128 and there are 8 of them. 128 x 8 = 1024 for a total length of all branches on the tree.	6. 1024 (B)
7. Digital Electronics	7. 001 (D)

The digital circuit translates to: $\overline{(A + \overline{\overline{(A + B)}\,\overline{(B C)}})}\ C$ $= \overline{A}\ \overline{\overline{\overline{(A + B)}\,\overline{(B C)}}}\ C$ $= \overline{A}\ \overline{(A + B)}\,\overline{(B C)}\ C$ $= \overline{A}\ (\overline{A}\,\overline{B})\,(\overline{B} + \overline{C})\ C$ $= \overline{A}\,\overline{B}\ (\overline{B} + \overline{C})\ C$ $= \overline{A}\,\overline{B}\,C + \overline{A}\,\overline{B}\,\overline{C}\,C$ $= \overline{A}\ \overline{B}\ C$ which is TRUE if A = 0, B=0, and C=1.	

8. Digital Electronics

The circuit translates to: (A)($\square$(A, B, C) + (($\square$(A, B, C) + C)

Let X = $\square$ (A, B, C) The expression is now: A X + (X + C)

A	B	C	X	AX	X + C	AX + (X+C)
0	0	0	0	0	0	0
0	0	1	1	0	1	1
0	1	0	1	0	1	1
0	1	1	0	0	1	1
1	0	0	1	1	1	1
1	0	1	0	0	1	1
1	1	0	0	0	0	0
1	1	1	0	0	1	1

Therefore there are 6 triples that make the expression TRUE.

8. 6 (D)

9. Prefix-Infix-Postfix

3 5 - 0 4 M 4 2 A 6 R * * 8 5 3 4 + M 3 A / R

= (3 5 -) 0 4 M (4 2 A) (6 R) * * 8 5 (3 4 +) M 3 A / R

= (((-2 0 4 M) (3 1/6 *) *) ((8 5 7 M) 3 A) /) R

= ((-2 1/2 *) (5 3 A) /) R

=(-1 4 /) R

= -1/4 R = -4

9. -4 (A)

10. Prefix-Infix-Postfix

+ * 2 ^ 3 - 3 5 / * 1 - + 3 5 1 ^ 3 2

= + * 2 ^ 3 (- 3 5) / * 1 - (+3 5) 1 (^ 3 2)

= + * 2 (^ 3 -2) / * 1 (- 8 1) 9

= + (* 2 1/9) / (* 1 7) 9

= + 2/9 (/ 7 9)

= (+ 2/9 7/9)   = 1

10. 1 (D)

11. Computer Number Systems

2020_{16} - 2020_{10} - 2020_{8}

$2020_{16} = 2 * 16^3 + 2 * 16 = 2 * 2^{12} + 32 = 2^{13} + 32 = 8192 + 32$

$= 8224_{10}$

$2020_8 = 2 * 8^3 + 2 * 8 = 2 * 2^9 + 16 = 2^{10} + 16 = 1024 + 16 = 1040_{10}$

$8224_{10} - 2020_{10} = 6204_{10} - 1040_{10} = 5164_{10}$

11. 5164 (B)

12. Computer Number Systems

$AB_{16} + DA_{16} = 185_{16} = 1\ 1000\ 0101_2 = 110\ 000\ 101_2 = 605_8$

$605_8 * 77_8 = 605_8 * (100_8 - 1_8) = 60500_8 - 605_8 = 57673_8$

$101\ 111\ 110\ 111\ 011_2$ which is 12 1's.

12. 12 (C)

13. Data Structures

The queue looks like C H R Y. Pop the C, then the H.

R Y S. Pop the R.

Y S A N T. Pop the Y.

S A N T H E. Pop the S and the A.

N T H E M U. Pop the N.

T H E M U M. Pop the T and the H.

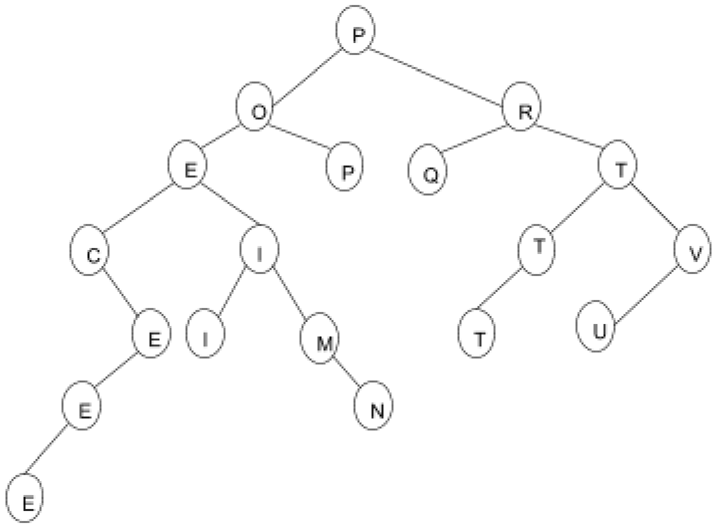
The longest length of the queue at any point in time is 6.

13. 6 (B)

14. Data Structures

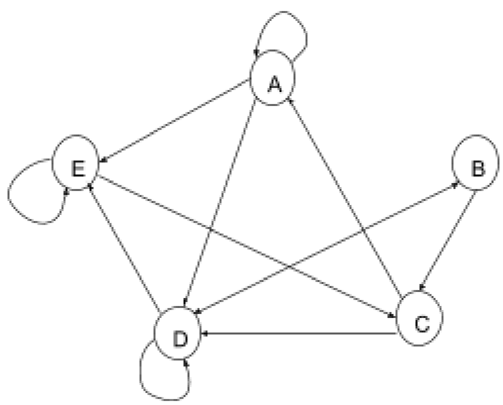
The binary search tree is drawn below. C, E, E, M, T, and V have only 1 child so there are 6 of them.

14. 6 (C)



15. Graph Theory

The graph from the adjacency matrix is below. By inspection, the cycles are: AA, DD, EE, BDB, AECA, ADBCA, ADECA, BCDB, and CDEC. There are 9 of them.



15. 9 (B)

16. Graph Theory

To find the number of paths of length 4, find the adjacency matrix, square it for paths of length 2, and square that for paths of length 4. By inspection, there are 5 values that are greater than 7.

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}^2 = \begin{bmatrix} 1 & 0 & 2 & 1 & 1 & 0 \\ 2 & 0 & 2 & 1 & 0 & 0 \\ 1 & 1 & 2 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 \\ 1 & 2 & 3 & 2 & 1 & 1 \end{bmatrix}^2 = \begin{bmatrix} 4 & 3 & 9 & 1 & 1 & 0 \\ 5 & 2 & 9 & 2 & 2 & 0 \\ 5 & 2 & 8 & 2 & 1 & 0 \\ 2 & 1 & 4 & 1 & 1 & 0 \\ 4 & 2 & 6 & 1 & 0 & 0 \\ 11 & 6 & 19 & 5 & 2 & 1 \end{bmatrix}$$

16. 5 (A)

17. What Does This Program Do?

17. 10 (C)

This program calculates the sum of the increasing factors of 2020 until that sum is more than 2020. They are 1, 2, 4, 5, 10, 20, 101, 202, 404 and 505.	
18. LISP Programming If Z = ‘(C (O N) (N (E C) T) (I (C (U) T))), S = ‘((N (E C) T) (I (C (U) T))) Y = (CAR (REVERSE S))) = (I (C (U) T)) X = (CAR (CDR ‘(C (U) T))) = (CAR ‘((U) T)) = (U) V = (CAR (CDR (CAR S))) = (CAR (CDR ‘(N (E C) T)))) = (CAR ‘((E C) T)) = (E C) (CONS V X) = ((E C) U)	18. ((E C) U) (B)
19. FSAs and Regular Expressions `[^aeiou]* [aeiou] [fghj-np-t] +. (ing|ful|age|less)?` a. brush|ing - OK b. help/ful - OK c. fractals - fails at C d. java - fails at V e. python! - OK f. shapeless - OK g. igloo - fails at second o h. apple - OK i. striving - fails at v j. image - fails at g k. grapple - OK l. rhythmic? - fails at c m. allstar - fails at second a n. syzygy - fails at end - no vowel	19. a, b, e, f, h, k (D)

o. covid - fails at v Therefore 6 of the choices satisfy the regular expression.	

20. Assembly Language
 	
The assembly programs can be converted to ACSL WDTPD code as follows:
 	
input n
while n != 0
    b = int(n / 10)
    x = b * 10
    c = n - x
    m = b + c
    y = m % 3
    if m == y then
        print n
    end if
    input n
end while
 	
This program checks if a given number is divisible by 3 by adding the digits to see if the sum is a multiple of 3. There are 4 such numbers before inputting 0: 24, 45, 51, 60.
	20. 4 (A)