American Computer Science League

2021 Finals ● Solutions to Short Problems ● Junior Division

1. Boolean Algebra

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

$$= 1(\overline{\overline{1}B})(\overline{B\overline{C}})$$

$$= \overline{B\overline{C}}$$

$$= \overline{B} \;+\; \overline{\overline{C}}$$

$$= \overline{B} \;+\; C$$

$\overline{B} \;+\; C$ is FALSE only for B = 1 and C = 0.
Therefore there are 3 ordered triples that make it TRUE:
(1, 0, 0), (1, 0, 1), and (1, 1, 1)

1. 3 (B)

2. Boolean Algebra

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

$$= \overline{0 \;+\; \overline{A}\,\overline{B} \;+\overline{A}\,B \;+\; B}$$

$$= \overline{\overline{A}(\overline{B} \;+\; B) \;+\; B}$$

$$= \overline{\overline{A} \;+\; B}$$

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

2. $A\,\overline{B}$ (C)

3. Bit-String Flicking

((LSHIFT-1 (NOT (RCIRC-2 01101) AND (LCIRC-2 01101)))
OR (RSHIFT-1 01101))
= ((LSHIFT-1 (NOT 01011 AND 10101)) OR 00110)
= ((LSHIFT-1 (10100 AND 10101)) OR 00110)
= ((LSHIFT-1 10100) OR 00110)
= (01000 OR 00110)
= 01110

3. 01110 (D)

4. Bit-String Flicking (NOT (01101 OR 01010) AND (01100 OR 10110)) = (NOT 01111 AND 11110) = (10000 AND 11110) = 10000	4. 10000 (D)
5. Recursive Functions $f(25) = 25 + f(25-3) = 25 + f(22) = 25 + 115 = 140$ $f(22) = 22 + f(22-3) = 22 + f(19) = 22 + 93 = 115$ $f(19) = 19 + f(19-3) = 19 + f(16) = 19 + 74 = 93$ $f(16) = 16 + f(16-3) = 16 + f(13) = 16 + 58 = 74$ $f(13) = 2 \cdot f(13+2) = 2 \cdot f(15) = 2 \cdot 29 = 58$ $f(15) = 15 + f(15-3) = 15 + f(12) = 15 + 14 = 29$ $f(12) = 12 + 2 = 14$	5. 140 (A)
6. Recursive Functions $f(5, 0) = 1$ $f(5, 1) = 5 \cdot f(5, 1-1) = 5 \cdot f(5, 0) = 5 \cdot 1 = 5$ $f(5, 2) = 5 \cdot f(5, 2-1) = 5 \cdot f(5, 1) = 5 \cdot 5 = 25$ $f(5, 3) = 5 \cdot f(5, 3-1) = 5 \cdot f(5, 2) = 5 \cdot 25 = 125$ $f(5, 4) = 5 \cdot f(5, 4-1) = 5 \cdot f(5, 3) = 5 \cdot 124 = 625$ $f(5, 5) = 5 \cdot f(5, 5-1) = 5 \cdot f(5, 4) = 5 \cdot 625 = 3125 > 1000$ Therefore $y = 5$. This function computes the powers of 5 recursively.	6. 5 (B)
7. Digital Electronics The boolean expression for this circuit is: $\overline{\overline{(AB)(B + C)} + C}$ $\overline{\overline{(AB)(B + C)} + C} = \overline{\overline{(AB)(B + C)}}\;\overline{C}$ $= (AB)(B + C)\overline{C}$ $= ABB\overline{C} + ABC\overline{C}$ $= AB\overline{C}$	7. $AB\overline{C}$ (D)

8. Digital Electronics

The boolean expression for this circuit is: $A + \overline{((A + B) + \overline{B\,C})}\,C$

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

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

$= A$ which is TRUE for (1, *, *) - 4 of them

8. 4 (C)

9. Prefix-Infix-Postfix

- + / + 1 8 3 * 4 3 ↑ 2 3 = - + / (+ 1 8) 3 (* 4 3) (↑ 2 3)
= - + (/ 9 3) 12 2^3
= - (+ 3 12) 8
= - 15 8
= 7

9. 7 (B)

10. Prefix-Infix-Postfix

$(x - h)^2 + (y - k)^2 = r^2$ ⇒ [(x - h)2↑] + [(y - k)2↑] = [r2↑]

⇒ [(x h - 2 ↑) + (y k - 2 ↑)] = [r 2 ↑]

⇒ [x h - 2 ↑ y k - 2 ↑ +] = [r 2 ↑]

⇒ x h - 2 ↑ y k - 2 ↑ + r 2 ↑ =

10. xh-2↑yk-2↑+r2↑= (A)

11. Computer Number Systems

$2021_{10} = 11111100101_2$

Add 2 more 1s: $11111101111_2 = 2031_{10}$

11. 2031 (A)

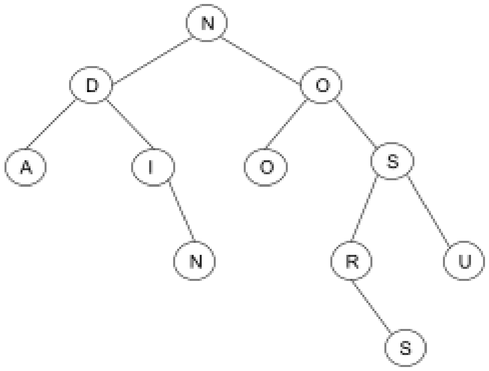
12. Computer Number Systems

$AB_{16} + 74_8 - 1101_2 = (10101011_2 + 111100_2) - 1101_2$
$= 11100111_2 - 1101_2$
$= 1101\ 1010_2$
$=$ D A $_{16}$

12. DA_{16} (B)

13. Data Structures

The binary search tree for NODINOSAURS is:



The depth of the tree is 4.

13. 4 (C)

14. Data Structures

A stack is LIFO. It is built as follows:

3	
3 7	
3 7 2	
3 7 2 4	
3 7 2	X = 4
3 7	Y = 2
3 7 6	X + Y = 6
3 7	X = 6
3	Y = 7
3 -1	X - Y = -1
3	X = -1
NIL	Y = 3
-3	X * Y = -3
-3 9	
-3 9 3	
-3 9	X = 3
-3	Y = 9
-3 3	Y / X = 3
-3	X = 3
NIL	Y = -3
-27	Y ^ X= -27

14. -27 (C)

15. Graph Theory

Squaring the adjacency matrix gives the number of paths of length 2.

$$\begin{vmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{vmatrix}^2 = \begin{vmatrix} 0 & 0 & 1 & 0 & 2 \\ 2 & 2 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 2 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{vmatrix}$$

Summing the entries yields 19 paths of length 2.

15. 19 (C)

16. Graph Theory

There are 8 cycles: ABCA, ABCDEA, ABEA, ADEA, ADEBCA, BCB, BCDEB, BEB.

16. 8 (B)

17. What Does This Program Do?

Choice A outputs multiples of 3 and outputs multiples of 5, but at 15 it outputs 15 twice (1 for each if).

Choice B outputs only multiples of 15.

Choice C outputs the multiples of 3 and outputs the multiples of 5.

Choice D outputs only multiples of 15.

17. if a(x,y) % 3 == 0 or
a(x,y) % 5 == 0 then
output(a(x,y)) (C)

18. What Does This Program Do?

The formula for the combination of n things taken r at a time is:

$${}_{n}C_{r} = \frac{n!}{r!(n-r)!}$$

The first loop calculates n!, the second loop calculates r! and the third loop calculates (n-r)!.

The output produces $\frac{8!}{3!5!}$. This computes to 56.

18. 56 (D)

19. What Does This Program Do?

This programs checks the elements in arr for an odd factor from 3 to half the number. This will determine if a number is prime. “check” will still be 1, i. e. 11, 43, 97. However this does not eliminate numbers that have only even factors like 16. That means 4 are outputted.

19. 4 (C)

20. What Does This Program Do?

len(s) = 11

So loop is from x = 5 to 0 step -1

When x = 5 the choices are A: s[0:5] or ACSL_F which eliminates A,
B: s[6:6] or I which eliminates B,
C: s[5:7] or FIN which eliminates C,
D: s[5:5] or F which is the first line.

Continuing the loop with D, x = 4 produces s[4:6] or _FI and then adds a letter each time on each end.

20. s[x : len(s) - x - 1] (D)