Zonal Informatics Olympiad, 2017

Instructions to candidates

1. The question paper carries 80 marks, broken up into four problems of 20 marks each. Each problem has three Test Cases. If you solve all three Test Cases correctly, you get 20 marks for that problem. Otherwise, you get 5 marks for each Test Case that you solve correctly.

2. All the $4 \times 3 = 12$ Test Cases appear as separate Questions in the right panel ("Question Palette").

   The first three Questions correspond to the three Test Cases of Problem 1, the next three correspond to the three Test Cases of Problem 2 and so on.

   A question icon turning green in the Question Palette, does not mean that it is correct. It just denotes that you have attempted it. All the questions will be evaluated later.

3. Attempt all questions. There are no optional questions.

4. There are no negative marks.

5. All expected answers are integers. Type in only the integer. So, if your answer is "162", enter only "162". Not "0162", or "162.0", etc.

6. Remember to save each answer. Only your final saved answers will be considered.

7. Near the top-right corner, you should be able to see a calculator icon. Clicking it pops up a calculator which you may use.

Zonal Informatics Olympiad, 2017

Questions

1. In this task you will give a sequence of numbers S of length N. Every element will be greater than or equal to 1. We write $S[i]$ to refer to the i^{th} element of the sequence. For example $S = (1, 2, 3, 4, 5, 4)$ is of length 6 and $S[2] = 2$, $S[6] = 4$ and so on. You will also be given a number T, called the target. We would like to find four elements of the sequence S whose sum is T. More precisely, we would like to find four numbers i, j, k, ℓ with $1 \leq i < j < k < \ell \leq N$ such that $T = S[i] + S[j] + S[k] + S[\ell]$.

   For instance with S as above, for $T = 13$, we can take $i = 2$, $j = 3$, $k = 4$ and $\ell = 6$. It can also be generated with $i = 1$, $j = 3$, $k = 4$ and $\ell = 5$ as well as $i = 1$, $j = 3$, $k = 5$ and $\ell = 6$. You can verify that these are only three ways to pick four positions summing upto 13.

   If $T = 16$ then there is a unique way to do this which is to take $i = 3$, $j = 4$, $k = 5$ and $\ell = 6$. Finally, $T = 9$ cannot be written as the sum of 4 elements of the sequence S and thus the number of ways in this case is 0.

   For the following S and T find the number of ways of expressing T as the sum of four elements of S.

   (a) $S = (2, 1, 1, 1, 2, 1, 2, 2, 1)$ and $T = 6$
   (b) $S = (2, 1, 2, 1, 3, 3, 1, 2, 1)$ and $T = 9$
   (c) $S = (1, 3, 2, 1, 1, 3, 4, 2, 2)$ and $T = 8$

2. A group of N people are on a hiking trip. It starts raining heavily and the group starts looking for places where they can be protected from the rain. There are 2 enclosures nearby where people can find shelter. Let's call them A and B. Enclosure A can accommodate X people, and Enclosure B can accommodate Y people. It is guaranteed that $(X + Y) \geq N$, and so everyone can be accommodated.

   The people are numbered from 1 to N. We know how far each person is from each enclosure. More specifically, for person i, $Adist[i]$ denotes how far the i^{th} person is from A and $Bdist[i]$ denotes how far the i^{th} person is from B.

   The group has genuine camaraderie, so they decide that they want to minimize the total distance traveled by their group members altogether to reach their respective enclosures. Note that this may mean that some individuals don't end up at the enclosure which is closest to them—the aim is to minimize the sum of the distances traveled by the members of the group.

   Given N, X, Y and the distances, your aim is to compute this minimum value.

   For example, if $N = 4$, $X = 2$, $Y = 2$, and the distances are

$$
\begin{aligned}
Adist[1] &= 10, & Bdist[1] &= 12\\
Adist[2] &= 23, & Bdist[2] &= 20\\
Adist[3] &= 15, & Bdist[3] &= 8\\
Adist[4] &= 5, & Bdist[4] &= 20
\end{aligned}
$$

Then the optimal strategy would be for 1 and 4 to go to A and for 2 and 3 to go to B. In this case the total distance traveled is

$$Adist[1] + Adist[4] + Bdist[2] + Bdist[3] = 10 + 5 + 20 + 8 = 43.$$

You can check that they cannot do any better.

Compute the corresponding minimum values for the following 3 inputs.

(a) $N = 8$, $X = 3$, $Y = 10$

$$\begin{array}{ll} Adist[1] = 4, & Bdist[1] = 6 \\ Adist[2] = 13, & Bdist[2] = 8 \\ Adist[3] = 12, & Bdist[3] = 21 \\ Adist[4] = 7, & Bdist[4] = 9 \\ Adist[5] = 51, & Bdist[5] = 37 \\ Adist[6] = 20, & Bdist[6] = 25 \\ Adist[7] = 5, & Bdist[7] = 17 \\ Adist[8] = 8, & Bdist[8] = 7 \end{array}$$

(b) $N = 12$, $X = 6$, $Y = 6$

$$\begin{array}{ll} Adist[1] = 10, & Bdist[1] = 4 \\ Adist[2] = 13, & Bdist[2] = 12 \\ Adist[3] = 13, & Bdist[3] = 27 \\ Adist[4] = 54, & Bdist[4] = 52 \\ Adist[5] = 6, & Bdist[5] = 9 \\ Adist[6] = 15, & Bdist[6] = 11 \\ Adist[7] = 21, & Bdist[7] = 26 \\ Adist[8] = 26, & Bdist[8] = 14 \\ Adist[9] = 37, & Bdist[9] = 33 \\ Adist[10] = 4, & Bdist[10] = 3 \\ Adist[11] = 11, & Bdist[11] = 11 \\ Adist[12] = 20, & Bdist[12] = 17 \end{array}$$

(c) $N = 20$, $X = 5$, $Y = 18$

$$\begin{array}{ll} Adist[1] = 3, & Bdist[1][B] = 1 \\ Adist[2] = 11, & Bdist[2] = 19 \\ Adist[3] = 20, & Bdist[3] = 24 \\ Adist[4] = 31, & Bdist[4] = 13 \\ Adist[5] = 72, & Bdist[5] = 102 \\ Adist[6] = 13, & Bdist[6] = 51 \\ Adist[7] = 36, & Bdist[7] = 38 \\ Adist[8] = 41, & Bdist[8] = 40 \\ Adist[9] = 4, & Bdist[9] = 9 \\ Adist[10] = 59, & Bdist[10] = 47 \\ Adist[11] = 19, & Bdist[11] = 21 \\ Adist[12] = 3, & Bdist[12] = 49 \\ Adist[13] = 28, & Bdist[13] = 32 \end{array}$$

$$
\begin{aligned}
Adist[14] &= 23, & Bdist[14] &= 29\\
Adist[15] &= 62, & Bdist[15] &= 62\\
Adist[16] &= 62, & Bdist[16] &= 59\\
Adist[17] &= 37, & Bdist[17] &= 34\\
Adist[18] &= 20, & Bdist[18] &= 20\\
Adist[19] &= 21, & Bdist[19] &= 17\\
Adist[20] &= 51, & Bdist[20] &= 54
\end{aligned}
$$

3. A group of N children, who are numbered $1, 2, \ldots, N$, want to play hide and seek. In a single round of hide and seek, there will one seeker, and $N-1$ hiders. Children like to hide and not seek and each child has her own idea of how many times she would like to hide. You will be given for each child i, a number $H[i]$ denoting the number of rounds she would like to be a hider. She will be satisfied only if she gets to be a hider in at least $H[i]$ rounds.

   For example, suppose $N = 4$, and $H[1] = 1$, $H[2] = 3$, $H[3] = 2$ and $H[4] = 1$. Here is one way to satisfy them all. In Round 1, Child 1 is the seeker, and in Rounds 2 and 3, Child 4 is the seeker. Then Child 1 has been a hider in 2 Rounds, Child 2 has been a hider in 3 Rounds, Child 3 has been a hider in 3 Rounds, and Child 4 has been a hider in 1 Round. Thus, they are all satisfied. You can check it is not possible to satisfy all of them in fewer than 3 Rounds.

   You aim is to determine the least number of rounds that needs to be played so that every child is satisfied. For the example in the previous paragraph, the answer is 3.

   Determine the minimum number of rounds needed in the following 3 cases:

   (a) $N = 7$
   $H[1] = 6$, $H[2] = 13$, $H[3] = 9$, $H[4] = 5$, $H[5] = 15$, $H[6] = 8$, $H[7] = 9$.

   (b) $N = 12$
   $H[1] = 6$, $H[2] = 7$, $H[3] = 7$, $H[4] = 8$, $H[5] = 9$, $H[6] = 9$, $H[7] = 9$, $H[8] = 9$, $H[9] = 9$, $H[10] = 9$, $H[11] = 9$, $H[12] = 9$.

   (c) $N = 15$
   $H[1] = 131$, $H[2] = 135$, $H[3] = 130$, $H[4] = 138$, $H[5] = 132$, $H[6] = 140$, $H[7] = 137$, $H[8] = 133$, $H[9] = 131$, $H[10] = 137$, $H[11] = 138$, $H[12] = 132$, $H[13] = 135$, $H[14] = 136$, $H[15] = 134$.

4. A binary string is a sequence of bits, i.e., a sequence 0s and 1s. For example, 01010 is a binary string of length 5.

   Suppose A and B are two binary strings. Then $A+B$ is the string which is obtained by pasting B at the end of A. This is called the concatenation of A with B. For example, if $A = 111$ and $B = 0000$na, then $A + B = 1110000$ and $A + A = 111111$.

   For a binary string A, A^k is A concatenated with itself k times. i.e. $A^k = \underbrace{A + A + \cdots + A}_{k \text{ times}}$. For example, if $A = 1011$, then $A^1 = 1011$, $A^2 = 10111011$ and $A^3 = 101110111011$.

   In this problem we are interested in binary strings of the form 0011^k. We call a binary string *Good* if it is 0011^k for some $k \geq 1$. For example, 0011 is Good, 00110011 is Good, 001100110011 is also Good and so on. But 001100111 is not Good. Neither is 11001100. Note that the empty string, i.e., the sequence with no bits, is not *Good*.

Suppose B is binary string. By dropping the bits in some of the positions in B we get a *substring* of B. For example, suppose $B = 0101100$. Then, dropping the bits in positions 2 and 6 results in the bit string 00110. Dropping the bits in positions 2 and 7 also yields the string 00110. On the other hand dropping the bits in positions 2, 4 and 5 yields the string 0000.

Given a binary string B, your aim is to find the number of ways of dropping some bits from B so that the resulting substring is Good.

For example, suppose $B = 10011$. Then we can drop the first bit and the resulting string is 0011 which is Good. You can verify that there is no other way to drop bits from B to get a Good string. The answer is hence 1. For the string 00011 there are 3 ways: drop the bit at position 1 to get 0011 or at position 2 to get 0011 or drop the bit at position 3 to get 0011. Hence, then answer in this case is 3.

Your task is to report the correct answer for the following 3 strings.

(a) $B = 001001101$

(b) $B = 100110101010011$

(c) $B = 01011000101001001$