F-Talent Code 2021 Programming Contest

Table A

F-Talent 2021

September 18, 2020

Problems

A. Project Schedule Management
B. Inventory Management
C. Sum of special series
E. Simple cryptography
G. Amateur Astronomer
H. 3 < 3 is Wrong Calculation
I. Moving Average
K. Pascal's Triangle Sum
L. Palindrome Level 2
M. BST duplicate
P. DNA

and others waiting for your reveal…

Do not open before the contest has started.

Advice, hints, and general information

- The problems are **not** sorted by difficulty.
- Your solution programs must read input from *standard input* (e.g. System.in in Java or cin in C++) and write output to *standard output* (e.g. System.out in Java or cout in C++). For further details and examples, please refer to your administrator guide and Domjudge documentation.
- For information about which compiler flags and versions are used, please refer to your administrator guide. (Python 2.7.17, Oracle Java 1.8.0_144, gcc 7.5.0 (C, C++ std14)).
- Your submissions will be run multiple times, on several different inputs. If your submission is incorrect, the error message you get will be the error exhibited on the first input on which you failed.
    - E.g., if your instance is prone to crash but also incorrect, your submission may be judged as either "Wrong Answer" or "Run Time Error", depending on which is discovered first. The inputs for a problem will always be tested in the same order.
- If you think some problem is ambiguous or underspecified, you may ask the judges for a clarification request through the Domjudge system. The most likely response is "No comment, read problem statement", indicating that the answer can be deduced by carefully reading the problem statement or by checking the sample test cases given in the problem, or that the answer to the question is simply irrelevant to solving the problem.
- In general we are lenient with small formatting errors in the output, in particular whitespace errors within reason, and upper/lower case errors are often (but not always) ignored. But not printing any spaces at all (e.g. missing the space in the string "1 2" so that it becomes "12") is typically not accepted. The safest way to get accepted is to follow the output format exactly.
- For problems with floating point output, we only require that your output is correct up to some error tolerance. For example, if the problem requires the output to be within either absolute or relative error of 10^{-4}, this means that
    - If the correct answer is 0.05, any answer between 0.0499 and .0501 will be accepted.
    - If the correct answer is 500, any answer between 499.95 and 500.05 will be accepted.

  Any reasonable format for floating point numbers is acceptable. For instance, "17.000000", "0.17e2", and "17" are all acceptable ways of formatting the number 17. For the definition of reasonable, please use your common sense.

A. Project Schedule Management

Time Limit: 3 seconds

Problem description

FPTU has a plan to build a new lecture hall named the Lambda at campus Hòa Lạc with the time schedule and dependencies of work items (preceding work) as described in the table format as below.

S.No	Work item	Period (days)	Preceding work

The following table is an example data:

S.No	**Work item**	**Period (days)**	**Preceding work**
1	A	3	-
2	B	5	A
3	C	3	A
4	D	10	B
5	E	7	B
6	F	4	C
7	G	5	E,F
8	H	2	D,G

- Line has S.No = 1 means that to finish work item named A, need at least 3 day, and no need to wait (no preceding work) to start.
- Line has S.No = 2 means that to finish work item named B, need at least 5 days, and to start B we need to wait until A finished at first.
- …
- Line has S.No = 7 means that to finish work item named G, need at least 5 days, and to start G we need to wait until both E and F finished before.

As a talent programmer, you are asked to program to determine ***the earliest possible time*** to complete project based on given data and an algorithm named CPM (Critical Path Method).

Input:

Line 1: an integer N which describes number of work items ($N<10^3$)

Line from 2 to N+1: each line describes the information of each work item, and has the following format: SNo WorkItem Period {PrecedingWork}

where the parts are separated by spaces

and the last part – Preceding work – described as a set of tasks that must be completed before, and enclosed in {}, each task separated by a comma. For example {A} hoặc {E,F} hoặc {w1,w3,w7}

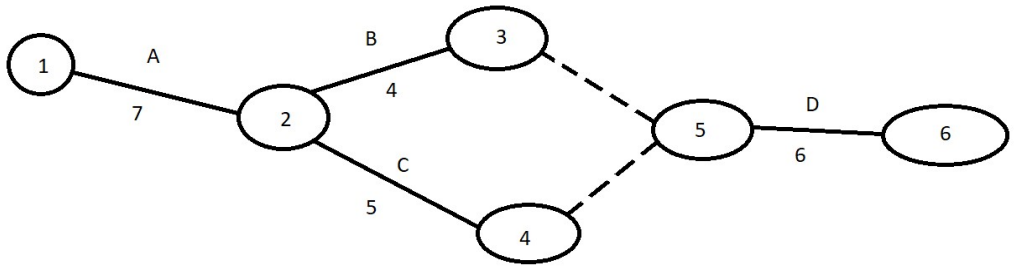
Output:

Output a single integer number which is the earliest possible time to complete project.

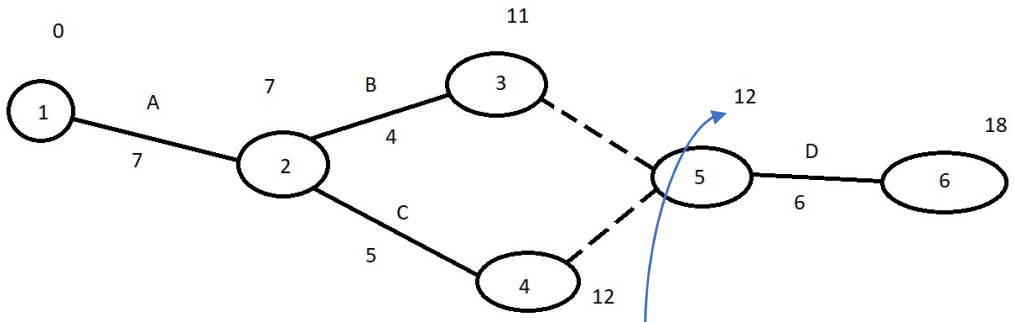
Example 1:

Input	Output
4 1 A 7 {-} 2 B 4 {A} 3 C 5 {A} 4 D 6 {B,C}	18

See illustrative graph below



After using CPM algorithm with forward calculation, we get the earliest possible time to complete each work item as the following graph.



Explain: To start work item named D, we need to finished both B and C before.

The earliest possible time to complete work item name B is 11 days

The earliest possible time to complete work item name C is 12 days

So, we get the greater value between 11 and 12 => 12 days