



Wagner

- Pending
- Questions
- Submissions
- Printouts
- Balloons
- Evolution
- Statistics
- Ranking

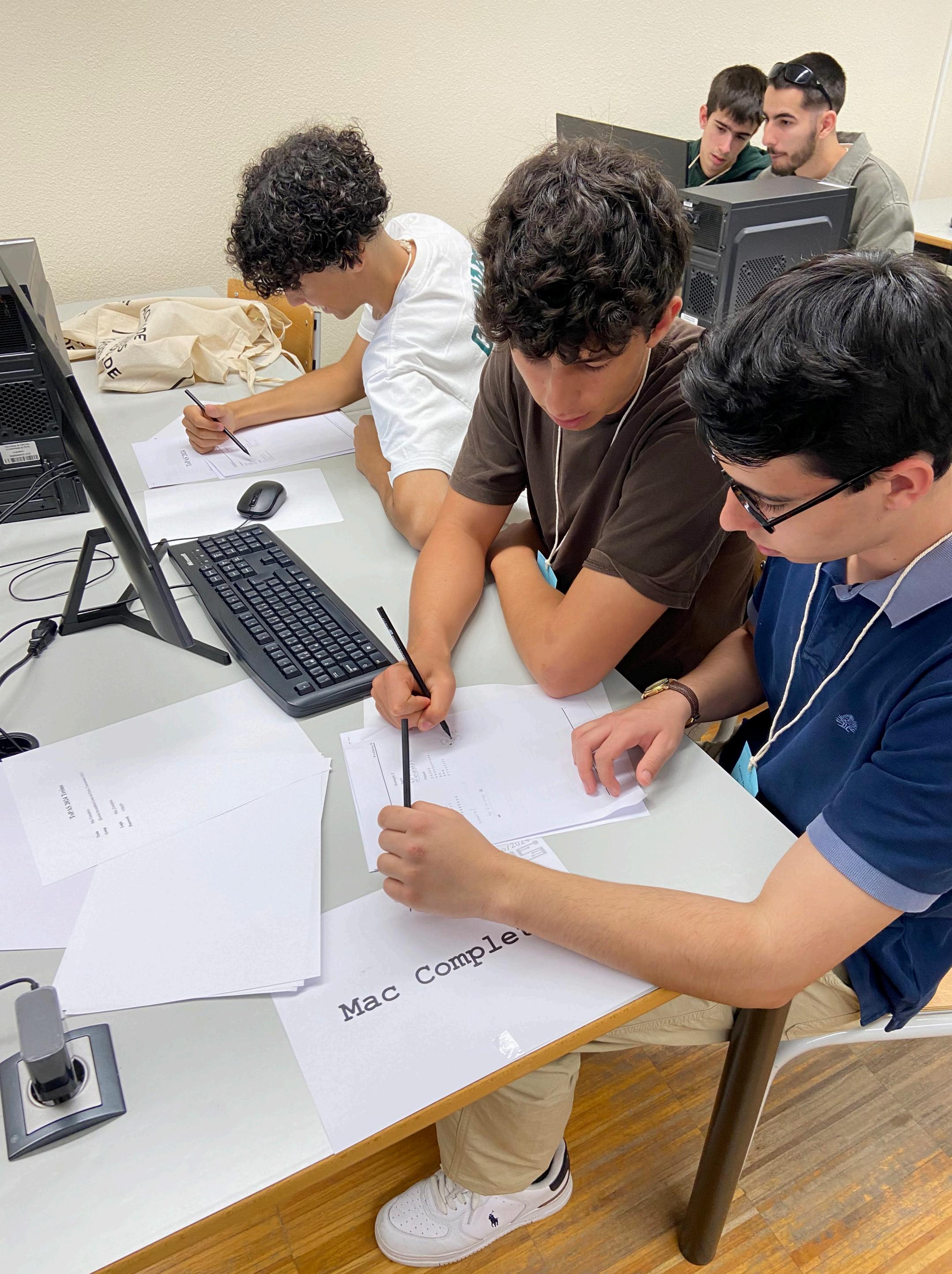
blems: Teams:

All Grouped

g1
123
16 MHz
AlgorithmToCode
AnittaMaxWynn
Belmiros
Blonde Brunette
COLGIA24
CarliTeams

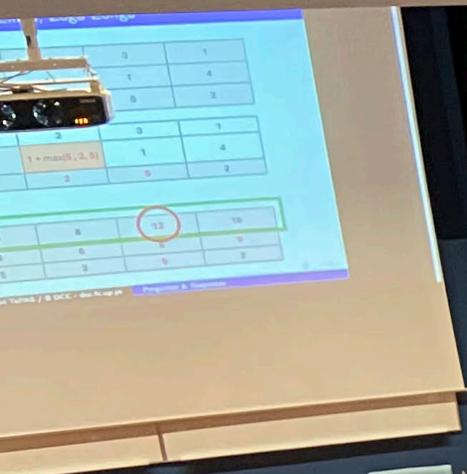
2022090807_0106_20
e0hanar qw.pdf





Mac Comple





FC FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO



#	Country	Team	Problems								Solved	Points
			A	B	C	D	E	F	G	H		
1		CIDC Belmiros	0:20:35 (1)	0:43:28 (1)	0:46:10 (2)	1:46:07 (1)	0:57:54 (0)	3:06:30 (4)			6	10:40:44
2		CIDC Blonde brunette	2:52:44 (3)	2:24:13 (3)	0:26:14 (0)	1:16:06 (0)	1:29:58 (0)	2:39:03 (1)			6	13:28:18
3		ESEN CodeAndCoffee	0:11:59 (0)	0:18:49 (0)	0:28:22 (0)	2:57:19 (3)		2:21:04 (2)			5	7:57:33
4		AESI AlgorithmToCode	0:09:10 (0)	0:43:59 (0)	1:09:23 (1)	1:04:31 (0)	----- (1)			----- (2)	4	3:27:03
5		AEF ProFaf	0:10:39 (0)	0:28:22 (1)	0:53:45 (0)			3:01:15 (2)			4	5:34:01
6		CGEC COLGAI24	0:22:55 (1)	0:14:24 (0)	0:29:18 (0)	2:54:58 (6)	----- (5)	----- (7)		----- (1)	4	6:21:35
7		CGEC Renegas'24	1:15:49 (2)	0:28:05 (0)	1:08:59 (2)	----- (3)	----- (1)	3:18:36 (2)		----- (1)	4	8:11:29
8		CIDC CodeWarriors	0:11:29 (0)	0:20:44 (0)	0:30:40 (0)	----- (2)		----- (3)			3	1:02:53
9		ESLC Mac Completo	0:10:30 (0)	0:30:53 (0)	1:36:53 (1)						3	2:38:16
10		AESI Eazy	0:57:23 (2)	0:50:08 (0)	1:40:02 (0)	----- (1)					3	4:07:33
11		AEOS JAG	1:52:04 (3)	0:54:22 (0)	1:18:30 (0)	----- (1)					3	5:12:55
12		CGEC CarliãoTeams	2:12:34 (3)	1:27:23 (0)	0:32:58 (0)			----- (2)		----- (2)	3	6:21:56
13		AE1G DIR	0:21:11 (0)	1:50:52 (1)	3:49:53 (0)						3	9:33:13
14		AESMF EZERU	2:26:41 (4)	3:15:21 (3)	1:31:11 (0)						2	0:49:59
15		AEPBS Joanenses	0:14:33 (0)	0:35:26 (0)	----- (6)	----- (1)					2	1:49:22
16		AEOS Guarda-chuva a vapor	----- (4)	0:49:50 (0)	0:59:32 (0)	----- (5)					2	2:16:20
17		AEPBS 123	0:18:19 (0)	1:18:01 (2)	----- (4)						2	4:38:28
18		ESTP 16 MHz	2:00:16 (0)		2:18:12 (1)	----- (4)					2	5:49:07
19		ESE Oreo Invertida	0:42:23 (0)	3:06:44 (6)	----- (3)						1	0:53:57
20		AESB .gll	0:33:57 (1)	----- (2)							1	1:09:52
21		AETP Três Ratos Cegos	----- (5)	1:09:52 (0)			----- (1)	----- (1)		----- (1)	1	1:16:03
22		AE1G YAKAIDYR	----- (3)	1:16:03 (0)	----- (1)						1	2:46:21
23		ESEN Debugful	----- (6)	2:26:21 (1)							0	0:00:00
24		AEC LIGAOLÉ	----- (7)	3:34:05 (6)	----- (1)						0	0:00:00
25		EPST SkyWhite	----- (31)	----- (2)	----- (1)						0	0:00:00
26		CL CyberSquad	----- (3)	----- (11)	----- (5)	----- (3)	----- (1)				0	0:00:00
27		EPM Os Invencíveis	----- (19)		----- (2)						0	0:00:00
28		AEOS The JAMS	----- (4)	----- (4)		----- (1)	----- (1)				0	0:00:00
29		AESI Rufus	----- (6)		----- (1)						0	0:00:00
30		AESB os OAZes	----- (3)	----- (1)	----- (2)						0	0:00:00
31		AE1G Rosário	----- (4)								0	0:00:00
32		EPM Isekai	----- (2)	----- (2)							0	0:00:00
33		ESTP JGL	----- (3)					----- (1)			0	0:00:00
34		ESE AnitaMaxWynn	----- (3)								0	0:00:00
35		EPST Preto e Branco	----- (1)								0	0:00:00
36		EPST Dois homens e melo									0	0:00:00

Dias até ao Natal

O Artur é um menino que sonha com o dia de Natal durante todo o ano. Ele imagina a casa repleta de risos e conversas animadas, o cheiro de bacalhau cozido pairando no ar e os sons suaves de músicas natalícias ecoando pela sala. Mas o que ele mais quer é ver os presentes debaixo da árvore para antecipar a sensação de felicidade de os desembrulhar.

Consegues ajudar o Artur escrevendo um programa que calcula quantos dias faltam até ao dia 25 de dezembro?



Tarefa

Escreva um programa que lê uma data, representada pelo dia e pelo mês do ano, e calcula o número de dias que faltam até ao próximo 25 de dezembro.

Vamos assumir que os anos em causa **não são bissextos**, isto é, que o mês de fevereiro tem 28 dias. Logo, o número de dias de cada mês é dado pela seguinte tabela:

Mês	1	2	3	4	5	6	7	8	9	10	11	12
Número de dias	31	28	31	30	31	30	31	31	30	31	30	31

Note que, se a data for depois de 25 de dezembro, o próximo Natal será no ano seguinte!

Input

A entrada consiste numa linha com dois inteiros, D e M , separados por um espaço, que representam, respetivamente, o dia e o mês de uma data.

Restrições

- $1 \leq M \leq 12$ O mês
- $1 \leq D \leq 28$ O dia se $M = 2$ (fevereiro)
- $1 \leq D \leq 30$ O dia se $M \in \{4, 6, 9, 11\}$ (meses com 30 dias)
- $1 \leq D \leq 31$ O dia se $M \in \{1, 3, 5, 7, 8, 10, 12\}$ (meses com 31 dias)

Output

A saída deve ter um único inteiro de 0 a 364, que representa o número de dias até ao próximo Natal.

Não metas água!

Imaginemos que temos de medir $1L$ (um litro) de água, mas só temos dois jarros, um de $3L$ e outro de $2L$. Uma solução será encher o jarro de $3L$ de uma torneira, depois o de $2L$ com a água do primeiro, que ficará exatamente com $1L$. Mas se tivermos um jarro de $5L$ e outro de $2L$? Podemos encher o de $5L$ da torneira, encher o de $2L$ com esta água, esvaziar esse jarro na pia, e voltar a enchê-lo com os $3L$ remanescentes do jarro maior, que fica apenas com $1L$.

Estes *puzzles*, que requerem medir volumes usando vários jarros não graduados, são um clássico em Inteligência Artificial. Podemos recorrer apenas a operações de três tipos: encher completamente um jarro a partir de uma torneira; esvaziar completamente um jarro para uma pia; ou encher um jarro com o conteúdo de outro. Neste último caso, ou o jarro que recebe a água fica completamente cheio, ou o jarro que fornece a água fica completamente vazio.

No final de uma sequência de operações destas, um dos jarros deverá ter um determinado volume alvo. Estes volumes, quer o alvo, quer a capacidade dos jarros, costumam ser valores inteiros positivos, todos na mesma unidade (litro, por exemplo).



Tarefa

Escreva um programa que recebe a descrição de um *puzzle* e uma sequência de operações, e determina se essa sequência é uma solução do *puzzle*, ou não.

Input

O input começa com a descrição do *puzzle*, que é seguida por uma sequência de operações.

A descrição do *puzzle* começa com $N + 1$ linhas, cada uma com um único inteiro. As primeiras N linhas têm as capacidades C_i dos N jarros, ou seja, o número na i -ésima linha representa a capacidade do jarro número i (para $i = 1, 2, \dots, N$). A última destas linhas tem o valor 0 . Segue-se uma linha com um inteiro A , que é o alvo a atingir.

Finalmente, as operações são descritas por S linhas, cada uma com dois inteiros, F e D , em que F representa a fonte da água e D o destino dessa água, com o seguinte significado:

- $F = 0$ a fonte da água é a torneira;
- $0 < F \leq N$ a fonte da água é o jarro número F ;
- $D = 0$ o destino da água é a pia;
- $0 < D \leq N$ o destino da água é o jarro número D .

Esta sequência termina com uma linha com dois zeros $(0\ 0)$.

Vai um?

Segundo o *Programa e Metas Curriculares para Matemática do Ensino Básico*, de 2013, é fundamental que os alunos adquiram “fluência de cálculo e destreza na aplicação dos quatro algoritmos, próprios do sistema decimal”, associados às quatro operações sobre os números naturais. Aprender a somar parece ter muito que se lhe diga. As metas para o 1º Ano referem “Adições cuja soma seja inferior a 100”, para o 2º Ano, “Adições cuja soma seja inferior a 1000” e, para o 3º Ano, “Algoritmos da adição e da subtração envolvendo números até um milhão”.



<https://www.youtube.com/watch?v=Twe2Hmg6ghM>

Queremos simplesmente explorar uma das várias estratégias que se ensinam para a soma, a designada por “conta em pé”. Para $28 + 45 = ?$, começa-se pelas unidades, $8 + 5 = 13$. Como excede 9, fica 3 nas unidades, e “vai 1”, para a posição seguinte, ou seja, para as dezenas. Como $1 + 2 + 4 = 7$ não excede 9, já não vai nada para as centenas, e conclui-se que $28 + 45 = 73$.

Vamos agora tentar um exemplo maior, $5489649 + 759340613 = ?$, com a “conta em pé”.

	0	0	1	0	1	1	1	0	1	0	← o que vai?
			5	4	8	9	6	4	9		
+	7	5	9	3	4	0	6	1	3		
	7	6	4	8	3	0	2	6	2		

Por vezes, erramos a conta. Se admitirmos que nunca nos enganamos quando somamos números inferiores a 10, quantas vezes nos esquecemos do “e vai 1” no exemplo seguinte?

	1	1	0	0	0	0	0	1	0	← o que vai?
			5	9	7	9	6	4	9	
+		9	9	9	2	0	6	1	3	
	1	0	4	8	9	9	2	6	2	

Duas vezes: quando passamos de $6 + 6$ para $9 + 0$ e quando passamos de $9 + 9$ para $5 + 9$. Os restantes algarismos errados são consequência destes erros e, por isso, não contam.

Tarefa

Escreva um programa que, dados dois inteiros positivos x e y , e o resultado r que obtivemos para $x + y$, possivelmente errado, indique quantas vezes nos esquecemos do “e vai 1”.

Input

A primeira linha tem três inteiros positivos, m , n e p , separados por um espaço, que são o número de dígitos de x , de y e de r , respetivamente. Seguem-se três linhas, com m , n e p inteiros (de 0 a 9) separados por espaços, que são os dígitos de x , y e r , lidos da esquerda para a direita, segundo a representação habitual, sem zeros à esquerda.



YAKAIDYR

Rafael

UNIVERSIDADE
DE PORTO
ALGARVES

Mac Completo