Computer University Magway

Programming Contest

2016

- **There are 6 problems (A-F) to solve within 5 hours (300 minutes).**
- **Solve as many problems as you can, in an order of your choice.**
- **Use Java to program at your convenience for any problems.**

Contest Problem Set

Problem A	**Hex Code**
Problem B	**Frequencies**
Problem C	**Wacmain Numbers**
Problem D	**Repeating Characters**
Problem E	**Penny Game**
Problem F	**Counting Weekends Days**

19 May 2016

PROBLEM A – HEX CODE

In the movie *The Martian* (2015), astronaut Mark Watney, one of the crew members of Mission Ares III, was left behind on Mars due to an unexpected incident during the surface exploration on the planet Mars. The communication with Earth was quasi-inexistent. Fortunately, Mark Watney managed to establish a very simple way to communicate with NASA at the mission control base on Earth through hexadecimal codes.

Mark could receive one simple code at a time which he could detect as a hexadecimal digit or hex code including `0-9` and `A-F`. Then he could transform *a pair of hex digits* into *a character* since he knows how to look up the ASCII table. Fortunately enough he has an ASCII table at hand (who would miss an ASCII table on voyage to Mars!! ^_^).

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	`
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(	72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29	)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-_	63	3F	?	95	5F	_	127	7F	DEL

If Mark receives a code sequence "4869204D61726B", he can decode it into the message "Hi Mark". However, decoding the hex codes manually is very slow and inefficient. Therefore your task is to help Mark Watney write a program to decode these hex codes.

Input

The input contains several lines of hex codes only (0-9, A-F). Each line contains an *even number* of hex digits that you have to transform into a plain text message. One pair of hex digits corresponds to a single character. There are less than 250 hex digits in each line. The input contains several lines of strings. You do not know in advance how many lines there are but it is guaranteed that the input contains less than 1,000 lines.

Output

For each line of hex codes, print out the corresponding text message.

Sample Input	Sample Output
4869204D61726B 41726520796F7520616C6976653F	Hi Mark Are you alive?

PROBLEM B – VOWELS, CONSONANTS AND DIGITS FREQUENCIES

The English alphabet consists of 26 letters and digits consist of 10 digits. Five of these alphabets (a, e, i, o and u) are classified as vowels, the remaining 21 as consonants. Digits are 10 numbers (0, 1, 2, 3, 4, 5, 6, 7, 8 and 9).

In this problem you will be give a number of pieces of English text. Your task is to determine the frequency of vowel, consonants and digit, these are found in the piece and to display the answers sorted by frequency, highest frequency first. Where two answers are equally frequent, they are to be displayed in alphabetical order.

As you can see from the examples below, upper case and lower case letters are considered to be the same letter in this problem. Use lower case in your output. As you can see from the fourth example, a frequency of zero must still be displayed.

INPUT FORMAT

Each piece of text to be analyzed is on a separate line of the input file. Each line has at most 200 characters. A single # on a line indicates the end of input.

Sample Input
ACM Programming Contest 2016. There are 35 students in class1. 2015-2016 academic year. This is a book. #

OUTPUT FORMAT

Output for a problem must be on a single line. Each answer must be output in lower case, followed by a colon, followed by the frequency of that answer. There must be one space before the next letter and a dot at the end.

Sample Output
`consonants:15 vowels:6 digits:4.`
`consonants:15 vowels:8 digits:3.`
`digits:8 consonants:6 vowels:6.`
`consonants:6 vowels:5 digits:0.`

PROBLEM C – WACMAIN NUMBERS

In the supposedly uninhabited Wacmahara Desert, a tribe of unusual people has been discovered. The Wacmians have only 2 fingers and a thumb on each hand, and have invented their own numbering system. The digits they use and the symbols they use for digits are quite unusual, but anthropologists have been able to represent them as follows:

`%` represents `0`
`)` represents `1`
`~` represents `2`
`@` represents `3`
`?` represents `4`
`\` represents `5`
`$` represents `-1` (yes, they even have a negative digit)

As you may expect, their system is base 6 where each place value is 6 times the value to its right, as in the following examples:

`)@%` is $1*6^2+3*6+0 = 36+18+0 = 54$
`?$~~` is $4*6^3+(-1)*6^2+2*6+2 = 864-36+12+2 = 842$
`$~~` is $(-1)*6^2+2*6+2 = -36+12+2 = -22$

Your task is to take Wacmian numbers and represent them as standard base 10 numbers.

INPUT FORMAT

Input consists of Wacmian numbers, one per line. Each number consists of a sequence of 1 to 10 Wacmian digits. A single '#' on a line by itself indicates the end of input.

SAMPLE INPUT:

```
)@%
?$~~
$~~
%
#
```

OUTPUT FORMAT

Output will be the corresponding decimal numbers, one per line.

SAMPLE OUTPUT:

```
54
842
-22
0
```

PROBLEM D- REPEATING CHARS

For this problem, you will write a program that takes a string of characters, ***S***, and creates a new string of characters, ***T***, with each character repeated ***R*** times. That is, ***R*** copies of the first character of ***S***, followed by ***R*** copies of the second character of ***S***, and so on. Valid characters for ***S*** are the *QR Code* "*alphanumeric*" characters:

```
0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ$%*+-./:
```

Input

The first line of input contains a single integer ***P***, ($1 \leq P \leq 1000$), which is the number of data sets that follow. Each data set is a single line of input consisting of the data set number ***N***, followed by a space, followed by the repeat count ***R***, ($1 <= R <= 8$), followed by a space, followed by the string ***S***. The length of string ***S*** will always be at least one and no more than 20 characters. All the characters will be from the set of characters shown above.

Output

For each data set there is one line of output. It contains the data set number, ***N***, followed by a single space which is then followed by the new string ***T***, which is made of the first character in ***S*** repeated ***R*** times, then the second and third character repeated ***R+1***, ***R+2*** times respectively and so on.

Sample Input	Sample Output
3 1 3 ABC 2 5 ./A 3 4 #PH	 1 AAABBBBCCCCC 2 .....//////AAAAAAA 3 ####PPPPPHHHHHH

PROBLEM E- PENNY GAME

Penny's game is a sample game typically played by two players. One version of the game calls for each player to choose a unique two-coin sequence such as **HEADS TAILS HEADS (HTH)**. A fair coin is tossed sequentially some number of times until one of the two sequences appears. The player who chose the first sequence to appear wins the game.

For the problem, you will write a program that implements a variation on the Penny Game. You will read a sequence of 10 coin tosses and determine how many times each three-coin sequence appears. Obviously there are eight such three-coin sequences: **TTT, TTH, THT, THH, HTT, HTH, HHT** and **HHH**. Sequences may overlap. For example, if all 10 coin tosses are heads, then the sequence **HHH** appears 8 times.

Input

The first line of input contains a single integer ***P***, ($\mathbf{1} \leq \boldsymbol{P} \leq \mathbf{1000}$) which is the number of data sets that follow. Each data set consists of 2 lines. The first line contains the data set number ***N***, The second line contains the sequence of 10 coin tosses. Each toss is represented as an upper case **H** or an upper case **T**, for heads or tails, respectively. There will be no spaces on any input line.

Output

For each data set there is one line of output. It contains the data set number followed by a single space, followed by the number of occurrences of each three-coin sequence, in the order shown above, with a space between each one. There should be a total of 9 space separated decimal integers on each output line.

Sample Input	Sample Output
4 1 HHHHHHHHHH 2 TTTTTTTTTT 3 HHTTTHHTTT 4 HTHTHHHTHH	 1 0 0 0 0 0 0 0 8 2 8 0 0 0 0 0 0 0 3 2 1 0 1 2 0 2 0 4 0 0 1 2 0 3 1 1

PROBLEM F- COUNTING WEEKEND DAYS

The contestants probably don't know how eagerly problem-setters (The people who prepare problems for a programming contest) wait for the weekend to make problems that would terrorize contestants :-). So before a month begins, some problem-setters try to calculate the number of weekend days in that month and plans accordingly. Can you help them to calculate this?

There are seven days in a week namely Sunday (**SUN**), Monday (**MON**), Tuesday (**TUE**), Wednesday (**WED**), Thursday (**THU**), Friday (**FRI**) and Saturday (**SAT**). There are twelve months in a year, January (**JAN**), February (**FEB**), March (**MAR**), April (**APR**), May (**MAY**), June (**JUN**), July (**JUL**), August (**AUG**), September (**SEP**), October (**OCT**), November (**NOV**) and December (**DEC**). These months have **31**, **28**, **31**, **30**, **31**, **30**, **31**, **31**, **30**, **31**, **30** and **31** days respectively. In leap years, the month of February has **29** days. In the bracket the three letter code for each month and each day are shown. Unlike many countries of the world Friday (**FRI**) and Saturday (**SAT**) are considered weekend days in Bangladesh. Given a month and the name of the first day of that month, you will have to find out the total no of weekend days in that month.

Input

First line contains an integer **T** (**T≤100**) which denotes the number of test cases. The input for each set is given in a single line. This line contains two strings **MTH** and **DAY**, here **MTH** is the three digit code of the month and **DAY** is the three digit code for the name of the first day of that Month.

Output

For each line of input produce one line of output. It contains a single integer which denotes the number of weekend days (Fridays and Saturdays) in that month. You must do your calculation assuming that the year is not a leap- year.

Sample Input	Output for Sample Input
3 JAN SUN FEB SUN OCT THU	8 8 10

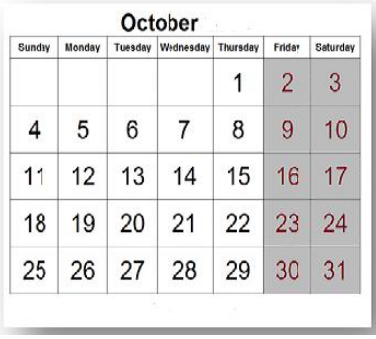

Illustration of third sample input:

In the third sample input we are asked to count the number of weekend days of a month October whose first day (October 1) is Thursday. The calendar on the left depicts this and it can be seen that there are 10 weekend days (colored red) in this month.

Problemsetter: Shahriar Manzoor, Special Thanks: Monirul Hasan, Muhammed Hedayetul Islam