Question **1**

Correct

Danny has a possible list of passwords of Manny's facebook account. All passwords length is odd. But Danny knows that Manny is a big fan of palindromes. So, his password and reverse of his password both should be in the list.

You have to print the length of Manny's password and it's middle character.

Note: The solution will be unique.

INPUT

The first line of input contains the integer N, the number of possible passwords.

Each of the following N lines contains a single word, its length being an odd number greater than 2 and lesser than ***14***. All characters are lowercase letters of the English alphabet.

OUTPUT

The first and only line of output must contain the length of the correct password and its central letter.

CONSTRAINTS

1 ≤ N ≤ 100

SAMPLE INPUT

4

abc

def

feg

cba

SAMPLE OUTPUT

3 b

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

int main() {
    int N;
    scanf("%d", &N);

    char words[110][20];
    for (int i = 0; i < N; i++) {
        scanf("%s", words[i]);
    }

    for (int i = 0; i < N; i++) {
        char rev[20];
        int len = strlen(words[i]);

        // Create reverse of words[i]
        for (int j = 0; j < len; j++) {
            rev[j] = words[i][len - 1 - j];
        }
        rev[len] = '\0';

        // Check if the reverse exists in the list
        for (int k = 0; k < N; k++) {
            if (strcmp(rev, words[k]) == 0) {
                // Found the correct password
                printf("%d %c\n", len, words[i][len / 2]);
                return 0;
            }
        }
    }

    return 0;
}
```

	Input	Expected	Got	
✓	4 abc def feg cba	3 b	3 b	✓
✓	59 kajworsgokugt bpwwaoqaigsyl ofjbnveyssqny hxtprvfqugfmi nsboestptjnxi iscvrluocokjg frntaenestypm sbatmloohhaex etiocpgpmjtan ptmhghijduusl ecojasfkhhcqy oavopnlcutldd rbakmfmbqouvy efbwgqwchopss mnfqswtvjgdxk tmhrlnyolfhkk paaispxaitycp iisgsouagiohx cmfmwwoaiipgk xckatskoagdwi lgysebilqhjfj rnyayydjbyeoo gutefgkfbqglc ytnfdfsfitgnf kmkybsvxwbgjg hapvdopsvubpc jljwtvjgunysl wtrgdbjuwoloh imiaornaohvbn sqnlemgxulyeg ystpbcejetesv uauvonoeasqiq lvsqerjxjkdow hkbaivbdurrjv tfmevaaoqeidc spfjmntppcosi lkawivcuabxar qebtivkofyebr vfipntapsjlvg	13 n	13 n	✓

	Input	Expected	Got	
	lyglqxrrrbper wgxaxtgglvcmm uxasgmrnalnuf fvvmpssntsjcb vxfkmaknvqcpg cbtwhcudokxja tckvfgwkqlmby itbbdgcwflbbq rlpbolfykmhhw xsmbuuwxxaeba kodlgvyxynjfo trvdrvyeytbwt ayydbjrfwxbxx aoihdeccivxja xfdwydxxiajbh ibkhamxkudoyf yugiwenadngbn odxrkgtwpvkni htwgywdfhalvc nbvhoanroaimi			

Passed all tests!

Question **2**

Correct

Joey loves to eat Pizza. But he is worried as the quality of pizza made by most of the restaurants is deteriorating. The last few pizzas ordered by him did not taste good ☹ . Joey is feeling extremely hungry and wants to eat pizza. But he is confused about the restaurant from where he should order. As always he asks Chandler for help.

Chandler suggests that Joey should give each restaurant some points, and then choose the restaurant having **maximum points**. If more than one restaurant has same points, Joey can choose the one with **lexicographically smallest** name.

Joey has assigned points to all the restaurants, but can't figure out which restaurant satisfies Chandler's criteria. Can you help him out?

Input:

First line has N, the total number of restaurants.

Next N lines contain Name of Restaurant and Points awarded by Joey, separated by a space. Restaurant name has **no spaces**, all lowercase letters and will not be more than 20 characters.

Output:

Print the name of the restaurant that Joey should choose.

Constraints:

$1 <= N <= 10^5$

$1 <= Points <= 10^6$

SAMPLE INPUT

3

Pizzeria 108

Dominos 145

Pizzapizza 49

SAMPLE OUTPUT

Dominos

Explanation

Dominos has maximum points.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

int main() {
    int N;
    scanf("%d", &N);

    char name[25], bestName[25];
    long long points, maxPoints = -1;

    for (int i = 0; i < N; i++) {
        scanf("%s %lld", name, &points);

        if (points > maxPoints) {
            maxPoints = points;
            strcpy(bestName, name);
        }
        else if (points == maxPoints) {
            if (strcmp(name, bestName) < 0) {
                // choose lexicographically smaller
                strcpy(bestName, name);
            }
        }
    }

    printf("%s", bestName);
    return 0;
}
```

	Input	Expected	Got	
✓	3 Pizzeria 108 Dominos 145 Pizzapizza 49	Dominos	Dominos	✓
✓	100 soghqhpbezpqthabaday 39 xbmcvrsidguczunngrw 80 fxkgwqlwwbjsvzuwaxac 48 alkvvhifvvmkxaruyqk 29 pmewilahjczknhilatcc 30 ggwuoezyljejduffuyyl 22 plvgiyuqpeqiwliduslt 81 fbnnjxrfqamxgcblsmm 52 wottstqcallpuhdqdkyk 85 xksasuewbxdynzgtsqa 42 serbvvpyflczsndmiwru 58 mujiqwkyvnwhrhppupbv 68 tylcqtljvquteggotrj 47 ltihchzljgqibamsasy 12 gzfibwlrorxudzywomlg 77 xsgwtwqmgllznpnmkpsp 56 psbiyikhkmzswivmzrs 48 kaaspswctmlhvmzlteqj 62 mtfncyrnxbpvqjmrmvuq 82 nsxrdmoppwxzwkolamj 74 ybuxgqfjpsycbjonzmy 93 tjndnhhsotfqtqgnhrqf 38 opuionioanpxibedzubp 94 kocdhbblezjfkgwvqvef 82 bpdfggnyweiudyajsiqq 40 arvdzfeeezawrxijcxc 72 gterlehneogppyigkhbh 71 ubrhvxbubnlcqhfkbun 47 xznesczsgjimpveudguo 34 lblloxkvzmprmwffimm 69 qgeofmzuhopvrdjsqhy 22 dhthdtysatyxywlbwit 65 daosaoiwjphfprvazuv 48 bturfecbiqhoyfvurei 44 gjvbjnbzqqarozismjx 96 ywfumkzqlbpkojfboxjq 64 euzplamwoljbwuchglw 4 uijrqdpocydtsbowujfj 27 dcwxhfnprhlrehapjqi 48 kyqbrxmikawadhcedhaa 13	dtzfqzqorwrpfgguxdi	dtzfqzqorwrpfgguxdi	✓

	Input	Expected	Got	
	omuievihgyasivtvnkmw 78			
	buqjacwdpwqgxpjnyju 81			
	ohkbewbeonngczniofpy 89			
	xokjkctolvrxkrcpleqr 14			
	slumsjyeyynzoienxlgv 6			
	abpwsexavswxbwyeavim 70			
	snlsbgfxexkvspscpuw 61			
	saulzzxcxhedtdjteyri 91			
	wekolxlrhpjncjzhrfe 51			
	hjekcrfgicibmzfsghkm 54			
	ofzuasqsqjltuaatwmat 66			
	uufxoorfzwxrobgokvvt 94			
	ltzvprjyzzhztsqueizy 80			
	dtzfqzqorwrpfgguxdi 99			
	bprozohkurwbcplmfno 37			
	xglnulbnvfodxxxabbct 33			
	wfbbtqchponfzwxjefts 94			
	joxoxskcmehoxbczbts 0			
	lnywajkivwalyneqrqf 13			
	ykllkrqidufqjqkkdxm 0			
	ntaswugdcluledzcudku 34			
	wooxkkgjfncshhydjsf 62			
	tcxrneltahtbgbdyjpv 33			
	iuhhlddmueexbvueykn 55			
	puqzkjsxiufjsiyywca 71			
	jaobrwxeinzsnycknur 95			
	feztboamfvvfpnaxpvx 94			
	hxilrkgtmmopyvmqgaak 95			
	gmmkcuhzoircwmcqniew 14			
	hsjqskvhwljnkfopkez 18			
	skkxsmmbeaycnmdqnefw 87			
	vnddasedcfvusgbxjti 89			
	snmevizbobwibcqukace 0			
	wvkhbhuilqzyaqyaiyab 21			
	iscujxhvlyfawephmitp 8			
	ihtsdvdrnrpnxobpsbzq 1			
	eucvvirxnawmqaopqprw 44			
	fzefjdzqwrxmygbthbdh 54			
	kjszjtfuseqrvbsyjwev 56			
	ndwtqepydnnuzaddcge 35			
	tjfomdpoaxufrnmunbvv 44			
	tzdcqscxqpphpyzoktlb 15			
	nlifscwwnkhqonudnnd 24			
	qxwaptrxobrqxrmivnb 15			
	krkvtgmueklwigfoeojn 54			
	bppqdmstztsiaybtyzvd 95			
	uofutultitxfalgxeigy 64			
	rjcqqfivxucflgqmuirq 8			

	Input	Expected	Got	
	yddkzgqakrcbqchvbjx 34 gqywgcdoghnhyhteaalf 43 mltunncgdjhrodappuv 77 iijqocykdxqrepgcqmi 22 ejvkpigdznwnirxscnb 47 ctdjjxobloiljcjlqoat 92 thmepijihqzkijqalaq 51 bqchnrujsrxvvgjnphcb 98 epcnxlxikmcgesevrqk 41 dkudbssgjtquykkjyslo 79 wcppdyrxyavwybibyax 60 tvrkdskzhpelyqyjqeg 69			

Passed all tests!

Question **3**

Correct

These days Bechan Chacha is depressed because his crush gave him list of mobile number some of them are valid and some of them are invalid. Bechan Chacha has special power that he can pick his crush number only if he has valid set of mobile numbers. Help him to determine the valid numbers.

You are given a string "S" and you have to determine whether it is Valid mobile number or not. Mobile number is valid only if it is of length 10 , consists of numeric values and it shouldn't have prefix zeroes.

Input:

First line of input is T representing total number of test cases.

Next T line each representing "S" as described in in problem statement.

Output:

Print "YES" if it is valid mobile number else print "NO".

Note: Quotes are for clarity.

Constraints:

$1 <= T <= 10^3$

sum of string length $<= 10^5$

SAMPLE INPUT

3

1234567890

0123456789

0123456.87

SAMPLE OUTPUT

YES

NO

NO

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int isValid(char s[])
{
    int len = strlen(s);

    // Condition 1: Must be exactly 10 characters
    if (len != 10)
        return 0;

    // Condition 2: Should not start with '0'
    if (s[0] == '0')
        return 0;

    // Condition 3: All characters must be digits
    for (int i = 0; i < len; i++)
    {
        if (!isdigit(s[i]))
            return 0;
    }

    return 1; // All conditions satisfied
}

int main()
{
    int T;
    scanf("%d", &T);

    while (T--)
    {
        char s[200];
        scanf("%s", s);

        if (isValid(s))
            printf("YES\n");
        else
            printf("NO\n");
    }
```

```
    }

    return 0;
}
```

	Input	Expected	Got	
✓	3 1234567890 0123456789 0123456.87	YES NO NO	YES NO NO	✓
✓	100 1234567890 326IG83A 21P5AA74 4PQDWXS9 67J914Y9 437P7MHB R130B7TC O1M7ZS2E DVD65L9A 7AZAF89I R9O9123Q 082287S0 UD9E0425 AYEN08H8 OG4O06AB 9R4823CN B454395P 25ZO7E10 A15JR677 G78ZFS6A 1E24HSK5 W9OQR9W7 5O6546E1 31HL1O42 43GR1845 37163VFJ 4NL5J835 L831H12K U51Y8006 4S8O9C41 15ME12BD ZJIAM45N Z237US34	YES NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO	YES NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO	✓

	Input	Expected	Got	
	09YN02PE	NO	NO	
	BL10KZ62	NO	NO	
	G9407ZM7	NO	NO	
	RI7O0MJ0	NO	NO	
	1A97XSEJ	NO	NO	
	XU10N952	NO	NO	
	HDT5ZP61	NO	NO	
	O68C043B	NO	NO	
	R0236590	NO	NO	
	717600F2	NO	NO	
	63B5YQ0B	NO	NO	
	Y15642FQ	NO	NO	
	3A6W013W	NO	NO	
	PP919N5Q	NO	NO	
	1DI7W75L	NO	NO	
	UT6022W5	NO	NO	
	645Q02P7	NO	NO	
	0XH3Y0TZ	NO	NO	
	404J26UK	NO	NO	
	8242CBI3	NO	NO	
	05AU7720	NO	NO	
	32747FP9	NO	NO	
	B9M462R7	NO	NO	
	5Y8ZZ935	NO	NO	
	H9W5CWS7	NO	NO	
	4D0UYW6E	NO	NO	
	09QI52OL	NO	NO	
	JEWB154R	NO	NO	
	252D7EN5	NO	NO	
	67L3PQW9	NO	NO	
	47343L62	NO	NO	
	Q96451CA	NO	NO	
	9XN4Q772	NO	NO	
	669X409A	NO	NO	
	4VZC59R9	NO	NO	
	1026RK71	NO	NO	
	29SZ6964	NO	NO	
	62PEEDQM	NO	NO	
	94S55400	NO	NO	
	2I1IR26J	NO	NO	
	5E790064	NO	NO	
	TTFN91TM	NO	NO	
	FF61631Y	NO	NO	
	14PCG58H	NO	NO	
	111V21S0	NO	NO	
	5NK23G4U	NO	NO	
	0QCG0OW6	NO	NO	
	9R4XH64F	NO	NO	

	Input	Expected	Got	
	T9829GD7	NO	NO	
	S0M0KKJ8	NO	NO	
	T778WE4H	NO	NO	
	N78G9462	NO	NO	
	3OVLVZ58	NO	NO	
	D8455C85	NO	NO	
	F3XL33T6	NO	NO	
	7ENQU5IJ	NO	NO	
	64H770D0	NO	NO	
	SO6MIYJ6	NO	NO	
	UWM9PJF6	NO	NO	
	W3Z78ZP6	NO	NO	
	7915O4DF	NO	NO	
	FGHUF04Q	NO	NO	
	087WL6O3	NO	NO	
	T3CJ7M93	NO	NO	
	U0V776RT	NO	NO	
	08KU7ZAJ	NO	NO	
	M111G1T3			

Passed all tests!