



Machine Translation by Homograph Detector with the Help of Grammatical Base of Persian Words

¹Prof. Dr. Zafer Agdelen; ²Dr. Amir Reza Shahbazkia

¹Rector, Bahcesehir Cyprus University, zafer.agdelen@cyprus.bau.edu.tr

²1105AmirReza@gmail.com

Abstract: Language is core medium of communication and translation is core tool for the understand the information in unknown language. Machine translation helps the people to understand the information of unknown language without the help of Human translator. This study is brief introduction to machine Translation and the solution for homographs. machine translation have been developed for many popular languages and many researches and developments have been applied to those languages but a significant problem in Persian (the language of Iranian, Afghani, etc.) is detecting the homographs which is not generally problematic in any other languages except Arabic. Detection of homographs in Arabic have been extensively studied. However Persian and Arabic share 28 characters, having only 4 different characters, they are two quite different languages. Homographs, words with same spelling and different translations are more problematic to detect in Persian because not all the pronounced vowels are written in the text (only 20% of vowels are written in the text) so the number of homographs in Persian is about thousands of times more than in other languages except Arabic.

In this paper we propose a new method for analysis and finding exact translation for homographs by algorithmic and grammatical rules.

Keywords: homograph disambiguation, machine translation, Statistical, homograph disambiguation

1. Introduction

significant problem in Persian (or Farsi) machine translation is homograph detection and disambiguation. This is not generally problematic in any other language except Arabic. Although a large work has been done for Arabic homograph detection and disambiguation with MADA [9], this work is useless for Persian. In fact Persian and Arabic are two quite different languages although they share 28 characters and have only 4 different ones. Since not all the vowels pronounced are actually written in the Persian and Arabic text, these two languages share a common problem in homograph detection and disambiguation but with different solutions.

Moreover the number of homographs in Persian is about thousands of times more than in other languages, except Arabic.

In Persian there are 32 characters from which 29 characters are consonants and the rest are vowels as shown below:

ب	Pronounced as b
پ	Pronounced as p
ت	Pronounced as t
ث	Pronounced as s
ج	Pronounced as j
چ	Pronounced as ch=C
ح	Pronounced as h
خ	Pronounced as kh=x

د	Pronounced as d
ذ ز ظ ض	Pronounced as z
ژ	Pronounced as zh=Z
ش	Pronounced as sh=S
ق غ	Pronounced as gh=q
ف	Pronounced as f
ک	Pronounced as k
گ	Pronounced as g
ل	Pronounced as l
م	Pronounced as m
ن	Pronounced as n
و	Pronounced as v
ی	Pronounced as y
آ ا ع	Pronounced as a (vowels)

From these 32 characters 28 characters are shared with Arabic and 4 are different: پ چ گ ژ pronounced as ch=C, p, zh=Z and g.

So one problem which arises in Persian texts which is not generally problematic in other languages such as English, Roman or Greek-based orthographies, is the identification of exact translation for homographs. For example the English sentence:

He prayed creator because he has sugar

Is written in Persian:

av (he) afrynnhdh (creator) ra () shkr (prayed) krd (did) kh (because) shkr (sugar) dard (has)

درد شکر که کرد شکر را آفریننده او

From right to left as written in Persian:

او آفریننده را شکر کرد که شکر دارد.

Pronounced in Persian:

Oo (he) afarirande (creator) raa () shokr (prayed) kard (did) ke (because) shekar(sugar) daard (has).

In the above example Skr (sugar) and Skr (prayed) are homographs, sugar is a noun and prayed is a verb so here we should distinguish between verb and noun to identify correct translation.

Even Google translator does not handle sentences with homographs very well.

This sentence is translated by site translate.google.com as:

I thank the creator that is.

For a better understanding of these translation examples consider that in the above sentence "ra", which has no correspondent English word, in Persian is the sign for object and is mostly used after the object. For example the English sentence:

I beat him.

Is written in Persian:

Mn (man) (I) av (oo) (him) ra (raa) () zdm (zadam) (beat). (in this sentence zdm is verb and av is object)

In Persian sentences sometimes (it just depends on the writer to choose whether to cancel the subject or not without any other criteria) we can cancel the subject by adding "m" (for I), "y" (for you), "" (for he, she and it), "ym" (for we), "yd" (for you) "nd" (for they) to end of verbs so in this sentence we can cancel "av".

But in the sentence:

He writes in a poor method

Written in Persian:

Av (he) bh (in) sbk (method) sbk (poor=low quality) mynvysd (writes)

او به سبک سبک مینویسد

As written from right to left:

او به سبک سبک مینویسد

Pronounced in Persian:

Oo(he) be(in) sabk(method) sabok(poor=low quality) minevisad (writes)

Is translated by site translate.google.com as:

His style of writing style.

As one sees sabk written as sbk is a noun and sabok with same spelling is an adjective so in this sentence identifying verbs is not the solution but identification between noun and adjective gives us the solution to understand the correct translation of these homographs.

A list of few examples for homographs is shown below:

byn as adjective pronounce bayyen means "good explainer" as noun pronounce beyn means "middle" as verb pronounce bin Means "look".

Bady as adjective pronounce badi means "next" as verb pronounce bodi means "you were"

sbk as adjective pronounce sabok means "light=low weight" as noun pronounce sabk means "style" as verb pronounce sabok means "to lightening"

mbyn as adjective pronounce mobin means "ownership" as noun pronounce mobayyen means "explainer" as verb pronounce mabin means "do not look at"

Bxvr as noun pronounce bukhur means "fumigation" as verb pronounce bokhor means "eat it"

Trkan as noun pronounce torkaan means “Turkish people” as verb pronounce tarakaan means “to explode”

Brdar as noun pronounce bordaar means “vector” as verb pronounce bardar means “take it”

rSt as noun pronounce rasht means “the name of city” as verb pronounce rosht means “growing up” (S stands for sh but s stands for s)

And hundreds of these words exist. As one can see many vowels are not written. In our solution we used an algorithm based on grammatical rules to reduce the search space in a database. The proposed algorithm tries to find not only verbs but nouns and adverbs as well.

2. Related Works

- 1) *Rule-based Machine Translation (RBMT) Approach:* The Rule-based Machine Translation works on the morphology, syntax and semantic of both languages. So, we required the syntax analysis, semantic analysis of Source text and to generate the text in target language we need syntax generation and semantic generation. We also need the bilingual dictionary of source and target languages. General Steps of Rule-based Machine Translation are described in figure 1.

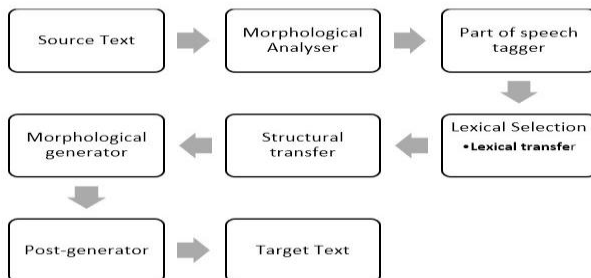


Fig. 1. Rule-based Machine Translation

Sub approaches in RBMT

The sub approaches in rule-based Machine translation are direct, transfer-based, interlingual Machine Translation approaches.

Direct Machine Translation Approach:

This is oldest approach and translation is performed at word level. There is no additional intermediary representation between source and target languages. Words of source language text is directly translated into the target language. This is uni-directional bilingual translation system. Direct machine translation approach involves the word by word translation with some modification at grammar level. The translation is not good as it is just the replacement of words from target language into source language text word by word meaning replacement.

Interlingual Machine Translation Approach:

This approach introduces an intermediary language representation between source and target languages. This

intermediary language is called Neutral Language. Neutral language can represent any natural language. It is independent of source and target Languages. It is also useful for multilingual translation machine system. KANT system was developed on interlingual approach in 1992 by Nyberg and Mitamura[10]. Building interlingual language is not an easy job. Too much efforts are required to develop truly neutral language.

Transfer base Machine Translation Approach In this approach the text of source language is converted into intermediary representation, it is then used to generate the target language text with help bilingual dictionary and grammar rules. Transfer based machine translation process is divided into three phases.

Analysis

In this phase source language text is analyzed on basis of linguistic information and heuristics to parser the text (syntactic representation)

Transfer

The syntactic representation of source language is converted into the syntactic form of target language.

Generation

The final text in target language is generate with help of morphological analysis. This approach heavily dependent on the grammar and structure of sentence and changes to a monolingual component affect all transfer modules for that language.

- 2) *Corpus-based Machine Translation Approach:* It is actually data driven machine translation. It was introduced an alternative approach to the rule-based approach. In this approach the bilanguage parallel corpus is used to extract the translation for new sentences. A large amount of raw data is collected in parallel corpora. The raw data is actually the translation between source and target languages and this data is used for translation. The sub-approaches of Corpus-based Machine Translation are Statistical Machine Translation and Example-based Machine Translation.

Statistical Machine Translation (SMT)

This approach is basis on statistical model. It has two statistical probabilities models: language model and translation model and massive parallel corpora of source and target languages. The advantage of SMT system is that linguistic knowledge is not required for building them. The difficulty in SMT system is creating massive parallel corpus. We have to two models in SMT, one is Word-based and other is phrase-based.

In word-bases MT sentences are consider as combination of single words and structure relation between the words are ignored while in phrase-based model consider sentences as combination of phrases or chunk. The basic concept in SMT is probability. The probability score of translations are generated from already available translated data (parallel corpus, translated by human), the translation having high probability is selected as final translation. The probability is calculated with help of language and translation models.

A huge amount of data is need for SMT and evolved many training repetition process. There is also no specific method quality control of corpora.

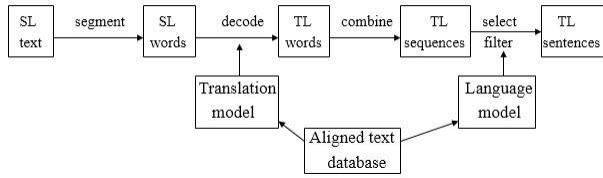


Fig. 2. Statistical Machine Translation

Example-based Machine Translation (EBMT)

Example-based machine translation contains the point to point mapping between the source and target language sentences i-e we have examples data that is translated between the source and target language [11]. This data is used for translation. The basic idea is if already translated sentence occur again it, the same translation is likely to be correct again. Basically, EBMT is memory-based translation and the concept of analogy is used for the translation.

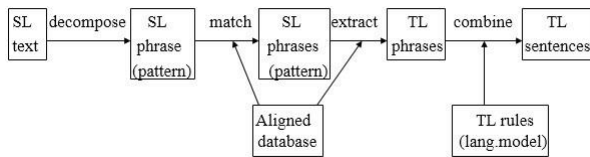


Fig. 3. Example-based Machine Translation

As one can realize in all previous works an important part does not exist and that part is homograph detector. We added this part in machine translation. Although this part added for Persian but it can be added to any other machine translators as a must.

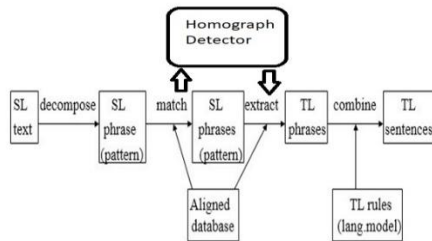


Fig. 4. Homograph Applied

Many studies on solving the above problem have been realized, such as a Work B- Like the Porter stemmer, for the English language [1], Work C-Persian stemmer algorithm works on the basis of the morphology of the language Afterwards Hessami Fard and Sani proposed a modified Krovetz Work D-algorithm for Persian stemming [2].

Work E-It uses POS tagging to increase performance and reduce errors to 60% Mokhtaripour and Jahnpour proposed a simple rule-based system for stemming Persian words [3].

Work F-In [4] Nasrin et al, presented a statistical stemmer for Persian text,

Work G- and Usefan et al, presented a study on the stemming challenges for Persian verbs and present on algorithm for Persian verbs [5].

Work I- verb detection in Persian corpus [7],

Work J- homographs in Persian morphology [6] and word sense disambiguation of Persian (Persian) homographs using thesaurus and corpus. Sense disambiguation of Persian (Persian) homographs using thesaurus and corpus. Currently, there is a lot of works discussing about machine translation but most of them apply statistical rules to use homographs. As a matter of fact homograph detection is not very important for other languages because the number of homographs is very low comparing to Persian. As explained before in this language vowels such as “a”, “e”, “o” and “u” is not written.

Hence the number of homographs in Persian is a huge number of words. In this field no work discussing Persian homograph detection completely; however, there are some works discussing about it. These works are mainly focused on verbs detection in sentences. For example work A, work E, work C, work D, work F, work G, work H, totally discussing about the subject but as a solution these can only detect the verbs in sentences.

In work J almost they have done a research to find verbs and more or less nouns but not adverbs and adjectives.

In the next figure we compare our research with other researches.

Fig. 6 shows data obtained out of 10,000 simple, 10,000 complex sentences, and 1000 traditional complex poems.

Works	Verb detector	Noun detector	Adverb Adjective detector
Work A	Yes	No	No
Work B	-	-	-
Work C	Yes	No	No
Work D	Yes	No	No
Work E	Yes	No	No
Work F	Yes	No	No
Work G	Yes	No	No
Work H	Yes	No	No
Work I	Yes	No	No
Work J	Yes	Yes	No
This research	Yes	Yes	Yes
Total	Normal	Normal	Good

Fig. 5. Compare Table Data obtained by 786 blind readers in 12 months duration.

Text	Simple sentences	Complex sentences	Traditional Poem
Verb detection	9,000	7,860	680
Noun detection	8,819	7,660	660
Adverb detection	7,755	6,619	566
Adjective detection	7,766	6,613	555
***** ****	***** ***	***** ***	***** ****
Verb detection	%90	%79	%68
Noun detection	%88	%77	%66
Adverb detection	%78	%66	%57
Adjective detection	%78	%66	%56
Total	%83.5	%72	%61.75

Fig. 6 Homograph detection

3. Discussion

In this part our algorithm is presented. First we substitute the Persian characters with Latin characters as listed above. Second we change it from right to left alignment to left to right alignment. Third in the text before word translation we detect homographs. Fourth we apply data base of thousands of homographs which made by dividing homographs in groups (verbs, nouns, adverbs, adjectives) then we present some grammatical based rules to identify the kind of homographs (noun, verb, adverb, adjective). At this point not only we apply homograph detector but also we apply statically rules. All names are divided in nine different groups N1-N9 such that popular names (book, apple, orange) goes in N1, the name of rivers, oceans, mountains, etc. in N2, people first names in N3, people surnames in N4, personal pronouns (I, you , he , she , we , they) in N6, (those, these, front, above, etc.) in N9 (although N6 and N9 does not hold nouns, they are processed as nouns) and N5, N7 and N8 are not currently Used. Verbs are divided in 4 different groups V1 to V3 plus VV, adverbs in 2 groups A0, A1 conjunctions in C0, prepositions in PR, and adjectives in Ad and word prefixes in Ab. After detecting the homograph is verb, noun, adjective, or adverb then we find exact meaning of it in the special meaning database. Finally we send the correct meaning of homograph to the rest of process of machine translation. Persian sentences are subject object verb, kind of sentences which means that the subject is always in the first position of a

sentence, then in the second position an object get place and the sentence is always ended by verbs. So if a word is the last word in a sentence the word is a verb.

To explain how to identify to which group the word belongs let's first observe the sentence as:

Word1 word2 word3 word4 word5 word6 word7 word8 word9 word10.

As we will see later identifying a word as a noun, a verb, an adverb or an adjective needs some proceeding and some preceding words to be considered. In this presentation it is assumed that the word next to word3 is word4 and the previous word is word2.

So each word in a sentence should be processed individually. If we are processing word(i) next word to this word is word(i+1) and previous word is word(i-1) , next word to the next word is word(i+2) and previous word to previous word is word(i-2).

This way a word can be identified according to some grammatical rules.

First of all the word (i) should be found in a group of Persian homographs then the following rules should be applied. It is worth mentioning that these rules are useful for homographs but not for all the words. Homographs identification is made by consulting a database with all the homographs, such as the ones in the appendix. Homograph translation is included in this database also.

- 1) If the word (i+1) is "v" (pronounced va means "and") or is "ta" (means "till") the word (i) is a verb.
- 2) If the word (i+1) is "ra" (sign of object in Persian) or "dr" (pronounced dar means "in") the word (i) is a noun.
- 3) If the word (i-1) can be found in group A1 the word (i) is a noun.
- 4) If the word (i-1) is a noun then the word (i) is an adverb or an adjective.

Please note that in Persian an adverb and an adjective cannot be homographs,

Since a word with some non-written vowels is pronounced the same way whether it is an adverb or an adjective.

Below it is presented an example of how this rule is applied.

- 5) If the word (i-1) is a conjunction or an adverb or a proposition then the word (i) is a noun.

- 6) If the word (i+1) is one of the words:

"bvd", "nbvd", "mbvd", "nmybvd", "bvdm", "bvdy", "bvdyd" or "bvdyd"

.then the word (i) is a noun

- 7) If the word (i+1) starts with one of the following part of words:

"nmygzar", "mygzar", "bgzar", "ngzar", "gzar", "krd", "grd", "nkrd", "mykrd", "kn", "bkn", "nkn", "mykn", "bzn", "nzn", "bvd"

.then the word (i) is a verb

- 8) If word (i+2) starts with "xvah" or "myxvah" then the word (i) is a verb.

The rules above are the main rules the algorithm uses. Some more auxiliary rules helps finding grammatical-base of homographs and hence the correct translation.

We can apply the rules above to find the correct translation of homographs in the following sentence:

Oo (written av) (he) afarirande (written afryndh) (creator) raa (written ra) (shokr (written Skr) (prayed) kard (written krd)(did)

ke (written kh) (because) shekar (written Skr) (sugar) daarad (written drd) (has).

In the above example Skr (sugar) and Skr (prayed) are homographs, sugar is a noun and prayed is a verb.

If we suppose shokr(written as Skr) is word(i) then word(i+1) is kard(written as krd) hence according to rule 7 Skr is a verb. If we suppose shekar (written as Skr) is word (i), then since word (i-1) is a preposition, according to rule 5 word (i) is a noun.

Let's use these rules again to identify verb and noun in the following sentence:

Bordaar (written brdar) (vector) raa (written ra) () bardaar (written brdar) (take)

In this sentence brdar (vector) and brdar (take) are homographs. According to rule 2 brdar (vector) is a noun and since brdar (take) is the last word in the sentence it is a verb.

The sentence will be the input. The words will be separated by the use of space between them. The number of words calculate. All of the words, search in the homograph database if even one word in the sentence is homograph then it be processed. The last word is verb.

```
if (number == lastnomre)
    search = "Vax"; // it means the last word is verb
```

The function SearchInWhat (...) as input word[i],word[i-1],word[i+1],word[i+2],...

The output of function is the grammatical value of the word[i]

It will be used for any single word in the sentence.

```
public string SearchInWhat(string nword, string nword,
string moute, string s2, string s3, int number, int lastnomre, int
lennword, string s0)
```

```
{
    string p, search;

    search = "Sax";
    if (number == lastnomre)
        search = "Vax";
    else
    {
        if (moute == "N6" || s0 == "on")
            search = "Nax";
        else
        {
            if (s2 == "ta" || s2 == "" || s2 == "v")
                search = "Vax";
            else
            {
                if (s2.Length < 3) s2 = s2 + "---";

                if (s2 == "ra" || moute == "A1" || s2 == "dr" ||
s2.Substring(0, 3) == "bvd")
                    search = "Nax";
                else
                {
                    if (s2 == ".")
                        search = "Vax";
                    else
                    {
                        if (s2.Length < 1)
                            search = "Vax";
                        else
```


(because we have the value of each word, which shows us the word is verb, noun or adverb). If the word is not homograph, hence it has only one specific value out of verb, noun or adverb.

The output of function "SearchInWhat" gives us whether the word which entered to this function, is verb, noun or adverb (adjective). After getting the result from this function, we search the word in specific database such as only verb database, noun database or adverb (adjective) database (Fig. 7.).

This process does for all words in one sentence. It tries to solve it like a puzzle as much as the words value (verb, noun or adverb) realized the puzzle solve easier. Hence this kind of process can solve complex sentences easier than any exist processes. A machine translator software for Persian according to the rules and regulations presented above was developed. This software was designed so that it can easily be tested. The feedback from testers was used to develop rules and to find any wrong rules. Then this software was distributed to blinds as a T.T.S (text to speech)

With Persian mother tongue. The algorithm was evaluated and tested over 2 years by 200 blinds to read different texts e.g. politics, social, economics, culture, art, religious and sport for about 2 hours per day. The reported results were acceptable.

Comparing this method with the ones proposed in [7], [9], [10] and [11] one can find that the method proposed in this paper can detect, besides verbs, also nouns adverbs and adjectives which are not supported by those methods.

Furthermore this method uses grammatical rules to reduce the search space in the database.

For illustrate, the sentence

"او به سبک سبک کار کرد و من را سبک کرد."

Equivalent letter changes to English and change it to left to write.

"av bh sbk sbk kar krd"

As it is realized there is not vowel written in Persian.

In the above example "sbk" is written three times as a verb, noun and adverb (adjective).

When one wants to read the above sentence. He/she will add

the vowels during the reading. It means this process take part in his/her brain.

"ou beh sabke sabok kar kard"

As it is very clear in this sentence nine vowels added. Hence in this language the number of homographs (because the vowels never written) is a huge number of words.

Unfortunately the previous works apply statistical approach to translate it [1],[2],[3],[4],[5]. And their result is.

"He worked in style"

It means the verb, adverb, adjective and noun is not detect clearly.

However some approaches in very simple sentences[6],[7],[8],[9] and [10] such as

"av bh sbk sbk mynvysd" Can detect the verb as mentioned before, but when the sentences become a little complex those cannot detect it like above example.

Our work translate the sentence "av bh sbk sbk kar krd" to "He worked in poor style." Unfortunately other approaches translate it to

"He worked in style"

As we know "sbk" as adjective pronounce "sabok" means "light=low weight" as noun pronounce "sab" means "style" as verb pronounce "sabok" means "to lightening".

All words in one sentence

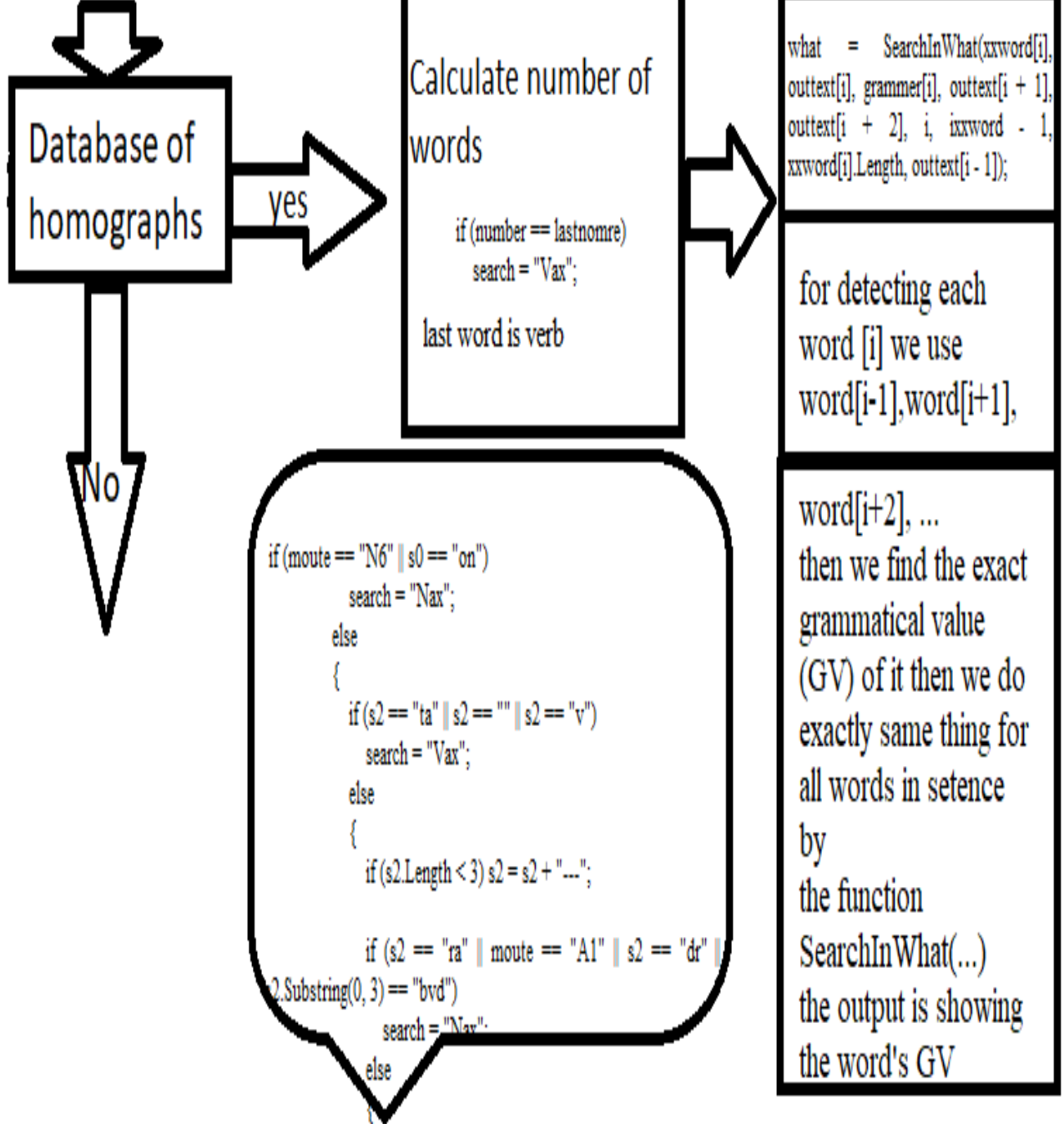


Fig. 7. The Process

4. Conclusion

In this paper is presented a new method to determine exact translation for Persian homographs, identifying verbs, based on grammatical rules. This is a new approach that we do not find in other algorithms like, multilingual translation machine system. KANT system was developed on interlingual approach in 1992 by Nyberg and Mitamura [10] Example-based machine translation contains the point to point mapping between the source and target language sentences i-e we have examples data that is translated between the source and target language [11] "A Stemming Algorithm for the Persian Language"[9] and "Verb detection in Persian corpus" [7].

Comparing to other methods, this approach has the advantage of not only detect verbs but also detect nouns, adverbs and adjectives. With acceptable accuracy.

References

- [1] M.F.Porter. An algorithm for suffix stripping Program 14 (3)pp 130-137, 1980
- [2] Reza Hessami Fard and Gholamreza Ghaem Sani. Stemmer algorithm design for Persian language. In Proceedings of the 11th international CSI computer conference (CSICC'2006) school of computer science IPM, Jan 24-26 2006, Tehran Iran.
- [3] Alireza Mokhtaripou and Saber Jahnpour. Introduction to a new Persian stemmer. In proceedings of CIKM'06, November 5-11 2006, Arlington, Virginia USA. ACM 1-59593-433 2/06/0011
- [4] Mojtaba Mohammad Nasrin, Kiyomars Sheikh Esmaeili and Hassan Abolhassani. A statistical stemmer for Persian language. In proceedings of the 11th international CSI computer conference (CSICC'2006) School of computer science IPM, Jan 24-26 2006, Tehran Iran
- [5] Ahmad Usefan, Somayeh Salehi and Behrouz Minaei Bidgoli. Stemming challenges and stemming algorithm for Persian verbs. In proceedings of the first workshop on Persian language and computers, Tehran University Iran May 25-26, 2004
- [6] M. Bijan khan and sh.Moradzadeh. Homographs in Persian Morphology. In proceeding of the first workshop on Persian language and computers, Tehran University, Iran, May 25-26, 2004
- [7] Majid Iranpour Mobarakeh and Behrouz Minaei Bidgoli. Verb Detection in Persian Corpus. JDCTA: International Journal of Digital Content Technology and its Applications, Vol. 3, No. 1, pp. 58-65, March 2009
- [8] Krovetz, R. (1993). Viewing morphology as an inference process. In Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 191-202.
- [9] CADIM Group. MADA+TOKAN: A System for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization [available online at: <http://www1.ccls.columbia.edu/~cadim/MADA.html>]
- [10] Eric H. Nyberg and Teruko Mitamura. The kant system: Fast, Accurate, high-quality translation in practical domains. In Proceedings of the 14th Conference on Computational Linguistics - Volume 3, COLING '92, pages 1069-1073, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics.
- [11] MD Okpor. Machine translation approaches: issues and Challenges. International Journal of Computer Science Issues (IJCSI), 11(5):159, 2014.

Appendixes

Source program and some example words

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Sina
{
    public partial class Form1 : Form
    {
        int telno = 0;
        string passwordx, xpassword="";
        string[] a = new string[70000];
        string[] b = new string[70000];
        string nword, outit;
        string voice = "F";
        int ixword,wavefile=0;
        string speedplus="0";
        string speedneg="0";

        public Form1()
        {
            InitializeComponent();
            password1();
            a[1] = "&k"; b[1] = "OhakNh";
            a[2] = "am"; b[2] = "OommNh";
            a[3] = "an"; b[3] = "OanNh";
            a[4] = "as"; b[4] = "OossNh";
            a[5] = "av"; b[5] = "OoNh";
            a[6] = "ax"; b[6] = "OexNh";
            a[7] = "b-"; b[7] = "ObaANh";
            a[8] = "bh"; b[8] = "ObenhNh";
            a[9] = "bl"; b[9] = "ObelNh";
            a[10] = "br"; b[10] = "ObarrNh";
            a[11] = "ck"; b[11] = "OcakNh";
            a[12] = "dh"; b[12] = "OdehNh";
            a[13] = "dk"; b[13] = "OdakNh";
            a[14] = "dm"; b[14] = "OdomNh";
            a[15] = "dr"; b[15] = "OdorrNh";
            a[16] = "dv"; b[16] = "OdoNh";
            a[17] = "dy"; b[17] = "OdeyNh";
            a[18] = "dz"; b[18] = "OdezNh";
            a[19] = "fk"; b[19] = "OfakkNh";
            a[20] = "fn"; b[20] = "OfanNh";
            a[21] = "fr"; b[21] = "OfarrNh";
            a[22] = "gh"; b[22] = "OgohNh";
            a[23] = "gl"; b[23] = "OgolNh";
            a[24] = "gr"; b[24] = "OgorNh";
```

```
a[25] = "hl"; b[25] = "OhelNh";
a[26] = "jd"; b[26] = "OjaddNh";
a[27] = "jr"; b[27] = "OjarrNh";
a[28] = "jv"; b[28] = "OjavvNh";
a[29] = "kh"; b[29] = "OkahNh";
a[30] = "kk"; b[30] = "OkakNh";
a[31] = "kl"; b[31] = "OkolNh";
a[32] = "km"; b[32] = "OkomNh";
a[33] = "kp"; b[33] = "OkopNh";
a[34] = "kr"; b[34] = "OkorNh";
a[35] = "ks"; b[35] = "OkasNh";
a[36] = "kt"; b[36] = "OkatNh";
a[37] = "ky"; b[37] = "OkiNh";
a[38] = "kz"; b[38] = "OkazNh";
a[39] = "lb"; b[39] = "OlabNh";
a[40] = "lf"; b[40] = "OlaffNh";
a[41] = "lh"; b[41] = "OlahNh";
a[42] = "lm"; b[42] = "OlemNh";
a[43] = "lp"; b[43] = "OlapNh";
a[44] = "md"; b[44] = "OmaddNh";
a[45] = "mh"; b[45] = "OmahNh";
a[46] = "ml"; b[46] = "OmelNh";
a[47] = "mn"; b[47] = "OmanNh";
a[48] = "mv"; b[48] = "OmovNh";
a[49] = "mw"; b[49] = "OmawNh";
a[50] = "my"; b[50] = "OmeyNh";
a[51] = "nh"; b[51] = "OnaNh";
a[52] = "ny"; b[52] = "OneyNh";
a[53] = "py"; b[53] = "OpeyNh";
a[54] = "pz"; b[54] = "OpozNh";
a[55] = "qd"; b[55] = "OqaddNh";
a[56] = "ql"; b[56] = "OqelNh";
a[57] = "qr"; b[57] = "OqerNh";
a[58] = "qv"; b[58] = "OqovNh";
a[59] = "rb"; b[59] = "OrabbNh";
a[60] = "rm"; b[60] = "OromNh";
a[61] = "rs"; b[61] = "OrosNh";
a[62] = "rv"; b[62] = "OroNh";
a[63] = "rx"; b[63] = "OraxNh";
a[64] = "rz"; b[64] = "OrazNh";
a[65] = "sm"; b[65] = "OsammNh";
a[66] = "sn"; b[66] = "OsenNh";
a[67] = "sr"; b[67] = "OsarNh";
a[68] = "ss"; b[68] = "OsesNh";
a[69] = "tk"; b[69] = "OtekNh";
a[70] = "tl"; b[70] = "OtallNh";
a[71] = "tn"; b[71] = "OtonNh";
a[72] = "tv"; b[72] = "OtoNh";
a[73] = "ty"; b[73] = "OteyNh";
a[74] = "vl"; b[74] = "OvolNh";
a[75] = "vt"; b[75] = "OvarNh";
a[76] = "vy"; b[76] = "OveyNh";
a[77] = "wk"; b[77] = "OwakkNh";
a[78] = "wl"; b[78] = "OwelNh";
a[79] = "wq"; b[79] = "OweqqNh";
a[80] = "wr"; b[80] = "OwerNh";
a[81] = "wv"; b[81] = "OwoNh";
a[82] = "ww"; b[82] = "OwewNh";
a[83] = "xm"; b[83] = "OxomNh";
a[84] = "xr"; b[84] = "OxerNh";
a[85] = "yd"; b[85] = "OyadNh";
```

a[86] = "yy"; b[86] = "OiNh";
a[87] = "zr"; b[87] = "OzarNh";
a[88] = "~y"; b[88] = "OteyyNh";
a[89] = "!hr"; b[89] = "OzahrNh";
a[51519] = "#daxlaqy"; b[51519] = "Ozeddeaxl1qiA0";
a[51520] = "#defvnyy"; b[51520] = "OzeddeofuniiX1";
a[51521] = "#dhvaayy"; b[51521] = "Ozeddehav1iiX1";
a[51522] = "#dhvayyy"; b[51522] = "Ozeddehav1yiiX1";
a[51523] = "#djasvyy"; b[51523] = "Ozeddej1susiA0";
a[51524] = "#dqanvny"; b[51524] = "Ozeddeq1noniA0";
a[51525] = "#dsr~any"; b[51525] = "Oseddesaret1niA0";
a[51526] = "#dtvlydy"; b[51526] = "OzeddetolidiA0";
a[51527] = "#dvyrvsy"; b[51527] = "OzeddevirusiA0";
a[51528] = "#d~alban"; b[51528] = "Ozeddet1leb1naA0";
a[51529] = "#d~bqaty"; b[51529] = "Ozeddetabeq1tiA0";
a[51530] = "#rabxanh"; b[51530] = "Ozarr1bx1neN1";
a[51531] = "#rabyany"; b[51531] = "Ozarr1biy1niXo";
a[51532] = "#radxanh"; b[51532] = "Ozarr1dx1neN1";
a[51533] = "#rbahngy"; b[51533] = "Ozarr1hangix1";
a[51534] = "#rvryaty"; b[51534] = "Ozaroriyy1tiX1";
a[51535] = "#yaaal&q"; b[51535] = "Oziy1olhaqqNa";
a[51536] = "\$&raayan"; b[51536] = "Osahr1iyInNo";
a[51537] = "\$&ranvrd"; b[51537] = "Osahr1navardA0";
a[51538] = "\$&ranwyn"; b[51538] = "Osahr1newinA0";
a[51539] = "\$&rapvry"; b[51539] = "Osahr1poriXo";
a[51540] = "\$&rayany"; b[51540] = "Osahr1iy1niXo";
a[51541] = "\$&rayyan"; b[51541] = "Osahr1iy1inNo";
a[51542] = "\$a&bdlyy"; b[51542] = "Os1hebdeliix1";
a[51543] = "\$a&bnry"; b[51543] = "Os1heb nazariN1";
a[51544] = "\$a&bqran"; b[51544] = "Os1he bqar1naA0";
a[51545] = "\$abvncy"; b[51545] = "Os1bonciiXo";
a[51546] = "\$adqyany"; b[51546] = "Os1deqiy1niXo";
a[51547] = "\$afkary"; b[51547] = "Os1fk1riiX1";
a[51548] = "\$al&yany"; b[51548] = "Os1lehiy1niXo";
a[51549] = "\$b&gahan"; b[51549] = "Osobhg1h1naP";
a[51550] = "\$ba^yany"; b[51550] = "Osabb1qiy1niXo";
a[51551] = "\$dabndyy"; b[51551] = "Osed1bandiiX1";
a[51552] = "\$dabrdar"; b[51552] = "Osed1bard1rn1";
a[51553] = "\$dabxwyy"; b[51553] = "OsadbaxwiiX1";
a[51554] = "\$dag@ary"; b[51554] = "Osed1goz1riiN1";
a[51555] = "\$dapyywhy"; b[51555] = "Osed1piweiX1";
a[51556] = "\$davsyma"; b[51556] = "Osed1osim1n1";
a[51557] = "\$ddrvazh"; b[51557] = "Osaddarv1zeNu";
a[51558] = "\$ddynary"; b[51558] = "Osaddin1riiN1";
a[51559] = "\$dqyany"; b[51559] = "Osedqiy1niiXo";
a[51560] = "\$drae!my"; b[51560] = "OsadreazamiN1";
a[51561] = "\$draldyn"; b[51561] = "OsadreddinNa";
a[51562] = "\$drnwyy"; b[51562] = "OsadrnewiniN1";
a[51563] = "\$drzadhy"; b[51563] = "Osadrz1deiXo";
a[51564] = "\$dsalgyy"; b[51564] = "Osads1legiiX1";
a[51565] = "\$dyqyany"; b[51565] = "Osed1qiy1niXo";
a[51566] = "\$faryany"; b[51566] = "Osaff1riy1niX1";
a[51567] = "\$farzadh"; b[51567] = "Osaff1rz1deNo";
a[51568] = "\$favrdyy"; b[51568] = "Osaff1verdiiXo";
a[51569] = "\$fdrxany"; b[51569] = "Osafdarx1niXa";
a[51570] = "\$fzradhy"; b[51570] = "Osafarz1deiXo";
a[51571] = "\$fvryany"; b[51571] = "Osaforiy1niXo";
a[51572] = "\$shyvnyism"; b[51572] = "OsahyunismN1";
a[51573] = "\$shyvnyst"; b[51573] = "OsahyunistA0";
a[51574] = "\$shyvnyzm"; b[51574] = "OsahyunizmN1";
a[51575] = "\$l&banyy"; b[51575] = "Osolhb1niiX1";

a[51576] = "\$smdzadhy"; b[51576] = "Osamadz1deiXo";
a[51577] = "\$ndvqchy"; b[51577] = "OsanduqceiX1";
a[51578] = "\$ndvqdar"; b[51578] = "Osandoqd1raA0";
a[51579] = "\$netgryy"; b[51579] = "OsanatgariiX1";
a[51580] = "\$netkary"; b[51580] = "Osanatki1riX1";
a[51581] = "\$rbstany"; b[51581] = "Oserbest1niXu";
a[51582] = "\$rdxzaey"; b[51582] = "Osoradexoz1iNa";
a[51583] = "\$vabdydy"; b[51583] = "Osav1bdidiX1";
a[51584] = "\$vltany"; b[51584] = "Osolatiy1niXo";
a[51585] = "\$vrtgryy"; b[51585] = "OsuratgariiX1";
a[51586] = "\$yadpvry"; b[51586] = "Osayy1dporiXo";
a[51587] = "\$yadyany"; b[51587] = "Osayy1diy1niXo";
a[51588] = "\$ydfvrwy"; b[51588] = "OseydforowiX1";
a[51589] = "%mrbxwyy"; b[51589] = "OsamarbaxwiiX1";
a[51590] = "%mvdgvnh"; b[51590] = "OsamudguneA0";
a[51591] = "%rvtmndy"; b[51591] = "OservatmandiN1";
a[51592] = "&\$arbdy"; b[51592] = "Ohes1rbandiN1";
a[51593] = "&\$arkwyy"; b[51593] = "Ohes1rkewiiX1";
a[51594] = "&\$rvra%t"; b[51594] = "Ohasrever1satN1";
a[51595] = "&\$yrbafy"; b[51595] = "Ohasirbi1fiN1";
a[51596] = "&a#r^ayb"; b[51596] = "Oh1zerq1yebN1";
a[51597] = "&a#rbawy"; b[51597] = "Oh1zerb1wiX1";
a[51598] = "&a#rjvab"; b[51598] = "Oh1zerjav1baA0";
a[51599] = "&a#ryraq"; b[51599] = "Oh1zeryar1qaA0";
a[51600] = "&a\$lxyzy"; b[51600] = "Oh1selxiziN1";
a[51601] = "&aaryany"; b[51601] = "Oh1eriy1niXo";
a[51602] = "&adsazy"; b[51602] = "Oh1ds1ziiX1";
a[51603] = "&ala&ala"; b[51603] = "Oh11h1l1Ap";
a[51604] = "&amdyany"; b[51604] = "Oh1mediy1niXo";
a[51605] = "&awavkla"; b[51605] = "Oh1w1vokall1Ap";
a[51606] = "&bybyany"; b[51606] = "Ohabibiy1niXo";
a[51607] = "&dadeadl"; b[51607] = "Oh1dd1del1delNo";
a[51608] = "&dadyany"; b[51608] = "Oh1dd1diy1niXo";
a[51609] = "&dadzadh"; b[51609] = "Ohadd1dz1deNo";
a[51610] = "&dgraayy"; b[51610] = "Ohadger1iiX1";
a[51611] = "&dgrayyy"; b[51611] = "Ohadger1yiiX1";
a[51612] = "&dv&saby"; b[51612] = "Ohaddohes1biX1";
a[51613] = "&jaryany"; b[51613] = "Ohajj1riy1niXo";
a[51614] = "&jazyany"; b[51614] = "Ohej1ziy1niXo";
a[51615] = "&kmfmay"; b[51615] = "Ohokmfarm1iX1";
a[51616] = "&kmrvayy"; b[51616] = "Ohokmrav1yiN1";
a[51617] = "&lvamahy"; b[51617] = "Ohalv1m1hiN1";
a[51618] = "&lvawkry"; b[51618] = "Ohalv1wekariN1";
a[51619] = "&lvawyrh"; b[51619] = "Ohalv1wireN1";
a[51620] = "&mdallhy"; b[51620] = "Ohamdoll1hiXa";
a[51621] = "&mdvsvrh"; b[51621] = "OhamdosoreN1";
a[51622] = "&mvrabyy"; b[51622] = "Ohamor1biiXa";
a[51623] = "&mydn-ad"; b[51623] = "Ohamidne71dNo";
a[51624] = "&mydr#ay"; b[51624] = "Ohamidrez1iXa";
a[51625] = "&mydyany"; b[51625] = "Ohamidiy1niXo";
a[51626] = "&nabndan"; b[51626] = "Ohan1band1nN1";
a[51627] = "&qartbar"; b[51627] = "Oheq1rtab1raA0";
a[51628] = "&qvqdany"; b[51628] = "Ohoquqd1niX1";
a[51629] = "&qyqyany"; b[51629] = "Ohaqiqiy1niX1";
a[51630] = "&ramxvar"; b[51630] = "Ohar1mx1rn1";
a[51631] = "&ramzadh"; b[51631] = "Ohar1lmz1deA0";
a[51632] = "&rvfcyny"; b[51632] = "OhurufciniX1";
a[51633] = "&sabdary"; b[51633] = "Ohes1bd1riiN1";
a[51634] = "&sabgryy"; b[51634] = "Ohes1bgariiX1";

a[51635] = "&sabrsyy"; b[51635] = "Ohes1bresiiX1";
a[51636] = "&sasytza"; b[51636] = "Ohass1siyyatz1A0";
a[51637] = "&sdrvzzy"; b[51637] = "OhasadvarziiX1";
a[51638] = "&synelyy"; b[51638] = "OhoseynaliiXa";
a[51639] = "&synqlyy"; b[51639] = "OhoseynqoliiXa";
a[51640] = "&synyany"; b[51640] = "Ohoseyniy1niXo";
a[51641] = "&wmdaryy"; b[51641] = "Ohawamd1riiX1";
a[51642] = "&ydra^ly"; b[51642] = "OheydarogliNa";
a[51643] = "&ydrbaba"; b[51643] = "Oheydarb1b1Na";
a[51644] = "&ydrelyy"; b[51644] = "OheydaraliiXa";
a[51645] = "&ydrn-ad"; b[51645] = "Oheydarne71dNo";
a[51646] = "&ydrpvry"; b[51646] = "OheydarpuriXo";
a[51647] = "&ydrxany"; b[51647] = "Oheydarx1niNo";
a[51648] = "&ydryyany"; b[51648] = "Oheydary1niXo";
a[51649] = "&ydrzadh"; b[51649] = "Oheydarz1deNo";
a[51650] = "&yvanyty"; b[51650] = "Oheyv1niyyatiX1";
a[51651] = "--vfyzyk"; b[51651] = "O7eofizikN1";
a[51652] = "--vlyyy"; b[51652] = "O7eolo7iiX1";
a[51653] = "--vwymyy"; b[51653] = "O7eowimiiX1";
a[51654] = "-a-dravy"; b[51654] = "O717dar1yiN1";
a[51655] = "-a-xayyy"; b[51655] = "O717x1yiiX1";
a[51656] = "-a-xvahy"; b[51656] = "O717x1hiN1";
a[51657] = "-andarky"; b[51657] = "O71nd1rkiXa";
a[51658] = "-andarmy"; b[51658] = "O71nd1rmiN1";
a[51659] = "-andarmry"; b[51659] = "O71nd1rmaniN1";
a[51660] = "-nratvry"; b[51660] = "O7ener1toriX1";
a[51661] = "-rfangry"; b[51661] = "O7arf1negariN1";
a[51662] = "-rfasnjy"; b[51662] = "O7arf1sanjiX1";
a[51663] = "-rfzyvyvy"; b[51663] = "O7arfziviX1";
a[51664] = " = "rmanymy"; b[51664] = " =
"O7erm1niyomiX1";
a[51665] = "-vlydgyy"; b[51665] = "O7olidegiiX1";
a[51666] = "-vrasaky"; b[51666] = "O7or1sikiX1";
a[51667] = "-vrdvzyy"; b[51667] = "O7ordoziiX1";
a[51668] = "-ybrlyny"; b[51668] = "O7iberliniX1";
a[51669] = "-ymnasty"; b[51669] = "O7imins1stiX1";
a[51670] = "-yrvskpy"; b[51670] = "O7iroskopiX1";
a[51671] = "-yvdivzyy"; b[51671] = "O7eodoziiX1";
a[51672] = "-yvfzyzyk"; b[51672] = "O7eofizikN1";
a[51673] = "-yvlv-yy"; b[51673] = "O7eolo7iiX1";
a[51674] = "-yvtaksy"; b[51674] = "O7eot1ksiN1";
a[51675] = "-yvwymyy"; b[51675] = "O7eowimiiX1";
a[51676] = "1-ydhaky"; b[51676] = "O17idh1kiXa";
a[51677] = "1-ydyaky"; b[51677] = "O17idiy1kiXa";
a[51678] = "11lvbalv"; b[51678] = "O1lob1loN1";
a[51679] = "1@araqyy"; b[51679] = "O1z1r1qiiX1";
a[51680] = " = "1@rgwnsb"; b[51680] = "
"O1zargownasbNa";
a[51681] = "1@rmydxt"; b[51681] = "O1zarmidoxNa";
a[51682] = "1@rmynay"; b[51682] = "O1zarmin1iXa";
a[51683] = "1@rn-ady"; b[51683] = "O1zarne71diXo";
a[51684] = "1@rstany"; b[51684] = "O1zarest1niX1";
a[51685] = "1@rsynay"; b[51685] = "O1zarsin1iXo";
a[51686] = "1@rwhayy"; b[51686] = "O1zarw1hiiXo";
a[51687] = "1^ajaryy"; b[51687] = "O1q1j1riiXu";
a[51688] = "1^azgryy"; b[51688] = "O1q1zgariiX1";
a[51689] = "1^azyany"; b[51689] = "O1q1ziy1niX1";
a[51690] = "1^dawlvy"; b[51690] = "O1qdl1wluXo";
a[51691] = "1badanyy"; b[51691] = "O1b1d1niiX1";
a[51692] = "1badgryy"; b[51692] = "O1b1dgariiX1";
a[51693] = "1badsazy"; b[51693] = "O1b1ds1ziN1";
a[51694] = "1bclykan"; b[51694] = "O1bcelik1nN1";
a[51695] = "1bdarcy"; b[51695] = "O1bd1rciiX1";
a[51696] = "1bdydggy"; b[51696] = "O1bdidegiiX1";
a[51697] = "1bjvsazy"; b[51697] = "O1bjos1ziN1";
a[51698] = "1bkvpyly"; b[51698] = "O1bkopiliX1";
a[51699] = "1bngaryy"; b[51699] = "O1bneg1riiX1";
a[51700] = "1brahamy"; b[51700] = "O1br1h1miXa";
a[51701] = "1brftgyy"; b[51701] = "O1braftegiiX1";
a[51702] = "1brsanyy"; b[51702] = "O1bres1niiX1";
a[51703] = "1brvbryy"; b[51703] = "O1berobariiX1";
a[51704] = "1brvdary"; b[51704] = "O1brod1riN1";
a[51705] = "1brvmndy"; b[51705] = "O1bromandiN1";
a[51706] = "1brvryzy"; b[51706] = "O1beruriziN1";
a[51707] = "1bryzgay"; b[51707] = "O1brizg1hiN1";
a[51708] = "1bsalany"; b[51708] = "O1bs1l1niX1";
a[51709] = "1bsrdkny"; b[51709] = "O1bsardkoniN1";
a[51710] = "1bvnmany"; b[51710] = "O1bunm1niX1";
a[51711] = "1bwvrany"; b[51711] = "O1bwor1niXu";
a[51712] = "1bwyryny"; b[51712] = "O1bwiriniX1";
a[51713] = "1bzdayyy"; b[51713] = "O1bzod1yiiX1";
a[51714] = "1bzydany"; b[51714] = "O1bzid1niX1";
a[51715] = "1bzyxanh"; b[51715] = "O1bzix1neN1";
a[51716] = "1carkwyy"; b[51716] = "O1c1rkewiiX1";
a[51717] = "1daptvry"; b[51717] = "O1d1ptoriX1";
a[51718] = "1dmyrally"; b[51718] = "O1dmir1liX1";
a[51719] = "1dmyzadh"; b[51719] = "O1damiz1deN1";
a[51720] = "1dmyzady"; b[51720] = "O1damiz1diX1";
a[51721] = "1dnvzyny"; b[51721] = "O1denoziniX1";
a[51722] = "1drnalyn"; b[51722] = "O1derna1linN1";
a[51723] = "1dnrn-yk"; b[51723] = "O1derner7ikN1";
a[51724] = "1dryatyk"; b[51724] = "O1driy1tikNu";
a[51725] = "1frydgar"; b[51725] = "O1farideg1rN1";
a[51726] = "1frykany"; b[51726] = "O1frik1niX1";
a[51727] = "1frynndh"; b[51727] = "O1farinandeA0";
a[51728] = "1frynwgr"; b[51728] = "O1farinewgarA0";
a[51729] = "1fryqaay"; b[51729] = "O1friaq1iA0";
a[51730] = "1fryqayy"; b[51730] = "O1friaq1yiA0";
a[51731] = "1ftabgvn"; b[51731] = "O1ft1bgonA0";
a[51732] = "1ftamaty"; b[51732] = "O1ft1ml1tiX1";
a[51733] = "1ftvmaty"; b[51733] = "O1ftom1tiX1";
a[51734] = "1gahandn"; b[51734] = "O1g1h1ndanVV";
a[51735] = "1gahanyd"; b[51735] = "O1g1h1nidVu";
a[51736] = "1gamnvny"; b[51736] = "O1g1mmoniXa";
a[51737] = "1gar1gar"; b[51737] = "O1g1r1g1rN1";
a[51738] = "1gnvstyky"; b[51738] = "O1gnostikN1";
a[51739] = "1grvwymy"; b[51739] = "O1growimiN1";
a[51740] = "1gvstyny"; b[51740] = "O1gostiniXa";
a[51741] = "1harmhrh"; b[51741] = "O1h1rmohreN1";
a[51742] = "1harznyy"; b[51742] = "O1h1rzaniiX1";
a[51743] = "1hngrany"; b[51743] = "O1hangar1niNo";
a[51744] = "1hngsazy"; b[51744] = "O1hangs1ziN1";
a[51745] = "1jrcynyy"; b[51745] = "O1jorciniX1";
a[51746] = "1jrkaryy"; b[51746] = "O1jork1riiX1";
a[51747] = "1jrnvwith"; b[51747] = "O1jornevewteN1";
a[51748] = "1jrparyy"; b[51748] = "O1jorp1reiX1";
a[51749] = "1jvdanyh"; b[51749] = "O1jod1niyyeNu";
a[51750] = "1jvdanyy"; b[51750] = "O1jud1niiX1";
a[51751] = "1jylyany"; b[51751] = "O1jilyi1niXo";
a[51752] = "1kadmyky"; b[51752] = "O1k1demikiA0";
a[51753] = "1kadymyk"; b[51753] = "O1k1demikA0";
a[51754] = "1karddvn"; b[51754] = "O1k1rdeonN1";

a[51755] = "1kardvny"; b[51755] = "O1k1rdeoniN1";
a[51756] = "1krvbasy"; b[51756] = "O1krob1siN1";
a[51757] = "1krvbaty"; b[51757] = "O1krob1tiA0";
a[51758] = "1krvpvly"; b[51758] = "O1kropolisNu";
a[51759] = "1krvpvly"; b[51759] = "O1kropoliXu";
a[51760] = "1krydyny"; b[51760] = "O1kridiniX1";
a[51761] = "1krylaty"; b[51761] = "O1kril1tiX1";
a[51762] = "1krylyky"; b[51762] = "O1krilikiX1";
a[51763] = "1ksfvrdy"; b[51763] = "O1ksfordiXu";
a[51764] = "1ktyndy"; b[51764] = "O1ktinidiX1";
a[51765] = "1ktynymy"; b[51765] = "O1ktiniyomiX1";
a[51766] = "1ktyvaay"; b[51766] = "O1ktiv1iN1";
a[51767] = "1ktyvyth"; b[51767] = "O1ktiviteN1";
a[51768] = "1kvarymy"; b[51768] = "O1kvIriyomiX1";
a[51769] = "1kvstyky"; b[51769] = "O1kostikiA0";
a[51770] = "1lagarsn"; b[51770] = "O111g1rsonA0";
a[51771] = "1lalgany"; b[51771] = "O111leg1niX1";
a[51772] = "1layndy"; b[51772] = "O111yandegiN1";
a[51773] = "1lbalvy"; b[51773] = "O11b1loyiA0";
a[51774] = "1lbatrvs"; b[51774] = "O11b1trosN1";
a[51775] = "1lbrkamv"; b[51775] = "O1lberki1moNa";
a[51776] = "1lbrtytm"; b[51776] = "O1lbertiniNa";
a[51777] = "1lbvmynh"; b[51777] = "O1lbomineA0";
a[51778] = "1lbvmyny"; b[51778] = "O1lbominiX1";
a[51779] = "1lbynysm"; b[51779] = "O1lbinismN1";
a[51780] = "1ldstrvn"; b[51780] = "O1ldosteronN1";
a[51781] = "1lgvnykn"; b[51781] = "O1lgonkiyanN1";
a[51782] = "1lgvrysm"; b[51782] = "O1lgorismN1";
a[51783] = "1lgvrytm"; b[51783] = "O1lgoritmN1";
a[51784] = "1lksandr"; b[51784] = "O1leks1nderNa";
a[51785] = "1lkmvtry"; b[51785] = "O1lkometriX1";
a[51786] = "1llymtry"; b[51786] = "O1llytimetriX1";
a[51787] = "1lvmyrysm"; b[51787] = "O1llomerismN1";
a[51788] = "1lvtrvpy"; b[51788] = "O1lotropiN1";
a[51789] = "1lwzadh"; b[51789] = "O1lewzi1deiXo";
a[51790] = "1lzaymy"; b[51790] = "O1lzi1lymeriX1";
a[51791] = "1lmadgahy"; b[51791] = "O1m1dg1hiX1";
a[51792] = "1lmajgahy"; b[51792] = "O1m1jg1hiX1";
a[51793] = "1malgamy"; b[51793] = "O1m1lg1miX1";
a[51794] = "1marbgay"; b[51794] = "O1m1rbegirA0";
a[51795] = "1margyry"; b[51795] = "O1m1rgiriN1";
a[51796] = "1marnamh"; b[51796] = "O1m1rn1meN1";
a[51797] = "1marwnas"; b[51797] = "O1m1rwn1sN1";
a[51798] = "1marylys"; b[51798] = "O1m1rilisN1";
a[51799] = "1mbvlans"; b[51799] = "O1mbol1nsN1";
a[51800] = "1mprdvry"; b[51800] = "O1mperdoriX1";
a[51801] = "1mprmtry"; b[51801] = "O1mpermetriX1";
a[51802] = "1mprsaet"; b[51802] = "O1mpers1atN1";
a[51803] = "1mprsnjy"; b[51803] = "O1mpersanj1X1";
a[51804] = "1mpyrysm"; b[51804] = "O1mpirismN1";
a[51805] = "1mrykaay"; b[51805] = "O1mr1k1iA0";
a[51806] = "1mrykayy"; b[51806] = "O1mr1k1yiA0";
a[51807] = "1mrykymy"; b[51807] = "O1merikiyomiX1";
a[51808] = "1mstrdam"; b[51808] = "O1mesterd1mNu";
a[51809] = "1mvdryay"; b[51809] = "O1mudary1iXu";
a[51810] = "1mvnyaky"; b[51810] = "O1moniy1kiA0";
a[51811] = "1mvxtggy"; b[51811] = "O1moxtegi1X1";
a[51812] = "1mvzanh"; b[51812] = "O1moz1neiX1";
a[51813] = "1mvzgary"; b[51813] = "O1muzeg1riN1";
a[51814] = "1mvzndgy"; b[51814] = "O1mozandegiN1";
a[51815] = "1mvzgwah"; b[51815] = "O1muzewg1hN1";
a[51816] = "1mvzwdh"; b[51816] = "O1mozewkadeN1";
a[51817] = "1mvzwyar"; b[51817] = "O1muzewy1rN1";
a[51818] = "1myxtggy"; b[51818] = "O1mixtegi1X1";
a[51819] = "1myzgary"; b[51819] = "O1mizg1riN1";
a[51820] = "1n-yvl-y"; b[51820] = "O1n7iyolo7iN1";
a[51821] = "1nablysm"; b[51821] = "O1n1bolismN1";
a[51822] = "1nahytay"; b[51822] = "O1n1hit1iXa";
a[51823] = "1nakavy"; b[51823] = "O1n1k1viiX1";
a[51824] = "1nakvnda"; b[51824] = "O1n1kond1N1";
a[51825] = "1narwysm"; b[51825] = "O1n1rwismN1";
a[51826] = "1narwyst"; b[51826] = "O1n1rwistN1";
a[51827] = "1natvlya"; b[51827] = "O1n1tolii1Nu";
a[51828] = "1natvlyy"; b[51828] = "O1n1toliiXu";
a[51829] = "1natvmyy"; b[51829] = "O1n1tomiiX1";
a[51830] = "1ndanth"; b[51830] = "O1nd1nte1X1";
a[51831] = "1ndrsvny"; b[51831] = "O1ndersoniXo";
a[51832] = "1ndrv-ny"; b[51832] = "O1ndero7eniX1";
a[51833] = "1ndvdrmy"; b[51833] = "O1ndodermiX1";
a[51834] = "1ndvgamy"; b[51834] = "O1ndog1miN1";
a[51835] = "1ndvkard"; b[51835] = "O1ndosmoziX1";
a[51836] = "1ndvkarp"; b[51836] = "O1ndok1rpN1";
a[51837] = "1ndvkrbn"; b[51837] = "O1ndokarbonN1";
a[51838] = "1ndvkryn"; b[51838] = "O1ndokrinA0";
a[51839] = "1ndvlnfy"; b[51839] = "O1ndolanfiX1";
a[51840] = "1ndvmtry"; b[51840] = "O1ndometriX1";
a[51841] = "1ndvskpy"; b[51841] = "O1ndoskopiN1";
a[51842] = "1ndvsmzy"; b[51842] = "O1ndosmoziX1";
a[51843] = "1ndvspm"; b[51843] = "O1ndospermN1";
a[51844] = "1ndvspvr"; b[51844] = "O1ndosporN1";
a[51845] = "1nglvfwb"; b[51845] = "O1ngelofobA0";
a[51846] = "1nglvfyl"; b[51846] = "O1ngelofilA0";
a[51847] = "1nglykan"; b[51847] = "O1ngelik1nA0";
a[51848] = "1ngstrmy"; b[51848] = "O1ngestromiX1";
a[51849] = "1ngvlayy"; b[51849] = "O1ngol1yiA0";
a[51850] = "1nkvlv-y"; b[51850] = "O1nkolo7iN1";
a[51851] = "1nmvmtry"; b[51851] = "O1nemometriX1";
a[51852] = "1nsfalyt"; b[51852] = "O1nsefi1litN1";
a[51853] = "1ntalpyy"; b[51853] = "O1nt1lpiiX1";
a[51854] = "1ntalyay"; b[51854] = "O1nt1liyiXu";
a[51855] = "1ntrakty"; b[51855] = "O1nter1ktiX1";
a[51856] = "1ntrasny"; b[51856] = "O1ntr1seniX1";
a[51857] = "1ntrasyt"; b[51857] = "O1ntr1sitN1";
a[51858] = "1ntrvkvk"; b[51858] = "O1ntrokokN1";
a[51859] = "1ntrvpyy"; b[51859] = "O1ntropiiX1";
a[51860] = "1ntvnyvy"; b[51860] = "O1ntoniyo1Xa";
a[51861] = "1ntymvan"; b[51861] = "O1ntimo1nN1";
a[51862] = "1ntymvny"; b[51862] = "O1ntimomiX1";
a[51863] = "1ntypaty"; b[51863] = "O1ntip1tiN1";
a[51864] = "1nvrysm"; b[51864] = "O1n1norismiX1";
a[51865] = "1nymysmy"; b[51865] = "O1nimismiX1";
a[51866] = "1pandysy"; b[51866] = "O1p1ndisiX1";
a[51867] = "1partayd"; b[51867] = "O1p1rt1yidN1";
a[51868] = "1partman"; b[51868] = "O1p1rtem1nN1";
a[51869] = "1praksyy"; b[51869] = "O1pr1ksiiX1";
a[51870] = "1pv1nzym"; b[51870] = "O1po1nzimN1";
a[51871] = "1qa\$fryy"; b[51871] = "O1q1safariiXo";
a[51872] = "1qaa&mdy"; b[51872] = "O1q1ahmadiNo";
a[51873] = "1qabxwyy"; b[51873] = "O1q1baxwiiXo";
a[51874] = "1qadadaw"; b[51874] = "O1q1d1d1wN1";

a[51875] = "1qadayy"; b[51875] = "O1q1d1yiiX1";
a[51876] = "1qajanyy"; b[51876] = "O1q1j1niiXo";
a[51877] = "1qakrymy"; b[51877] = "O1q1karimiNo";
a[51878] = "1qamvwhy"; b[51878] = "O1q1moweiX1";
a[51879] = "1qamyryy"; b[51879] = "O1q1miriiXo";
a[51880] = "1qamyrza"; b[51880] = "O1q1mirz1Na";
a[51881] = "1qana\$ry"; b[51881] = "O1q1n1seriNo";
a[51882] = "1qanbaty"; b[51882] = "O1q1nab1tiNo";
a[51883] = "1qanvazy"; b[51883] = "O1q1nav1ziXo";
a[51884] = "1qar#ayy"; b[51884] = "O1q1rez1yiNo";
a[51885] = "1qaxanyy"; b[51885] = "O1q1x1niiXo";
a[51886] = "1qazadhy"; b[51886] = "O1q1z1deiXo";
a[51887] = "1qvny~vn"; b[51887] = "O1qonitonN1";
a[51888] = "1r-antyn"; b[51888] = "O1r71ntinNu";
a[51889] = "1r-antyt"; b[51889] = "O1r71ntitN1";
a[51890] = "1rambndy"; b[51890] = "O1r1mbandiX1";
a[51891] = "1ramgahy"; b[51891] = "O1r1mg1hiA0";
a[51892] = "1ramgany"; b[51892] = "O1r1mg1niX1";
a[51893] = "1rastgyy"; b[51893] = "O1r1stegiiX1";
a[51894] = "1raywgah"; b[51894] = "O1r1yewg1hN1";
a[51895] = "1raywgyr"; b[51895] = "O1r1yewgariN1";
a[51896] = "1rdsazyk"; b[51896] = "O1rds1ziiX1";
a[51897] = "1rdvarky"; b[51897] = "O1rdv1rkiX1";
a[51898] = "1rmangra"; b[51898] = "O1rm1nger1A0";
a[51899] = "1rmatvry"; b[51899] = "O1rml1toriX1";
a[51900] = "1rmynyay"; b[51900] = "O1rmini1yiXa";
a[51901] = "1rmyta-y"; b[51901] = "O1rmit17iXa";
a[51902] = "1rvbndyy"; b[51902] = "O1robandiX1";
a[51903] = "1rvmatyk"; b[51903] = "O1rom1tikN1";
a[51904] = "1rwydvky"; b[51904] = "O1rwidokiX1";
a[51905] = "1rwydvws"; b[51905] = "O1rwidowesN1";
a[51906] = "1rwytkty"; b[51906] = "O1rwiktekiX1";
a[51907] = "1rwyvyst"; b[51907] = "O1rwiivistN1";
a[51908] = "1ryamnwy"; b[51908] = "O1riy1manewiXo";
a[51909] = "1ryan-ad"; b[51909] = "O1riy1ne71dA0";
a[51910] = "1ryanfry"; b[51910] = "O1riy1nfariXo";
a[51911] = "1ryanysm"; b[51911] = "O1riy1nismN1";
a[51912] = "1ryapvry"; b[51912] = "O1riy1poriXo";
a[51913] = "1ryazndy"; b[51913] = "O1riy1zandiXo";
a[51914] = "1ryvbrzn"; b[51914] = "O1ryobarzanNa";
a[51915] = "1ryzvny"; b[51915] = "O1rizon1iXu";
a[51916] = "1rzvanhy"; b[51916] = "O1rezov1neiX1";
a[51917] = "1rzvmndy"; b[51917] = "O1rezumandiN1";
a[51918] = "1sangvar"; b[51918] = "O1s1ngov1rA0";
a[51919] = "1sansvry"; b[51919] = "O1s1nsoriX1";
a[51920] = "1saywgah"; b[51920] = "O1s1yewg1hN1";
a[51921] = "1skarysy"; b[51921] = "O1sk1risiX1";
a[51922] = "1smankan"; b[51922] = "O1sem1nk1nN1";
a[51923] = "1spyryny"; b[51923] = "O1spiriniX1";
a[51924] = "1statyny"; b[51924] = "O1st1tiniX1";
a[51925] = "1strgyry"; b[51925] = "O1stargiriN1";
a[51926] = "1strvl-y"; b[51926] = "O1strolo7iN1";
a[51927] = "1strvydy"; b[51927] = "O1steroidN1";
a[51928] = "1strvydy"; b[51928] = "O1steroidiX1";
a[51929] = "1stygmat"; b[51929] = "O1stigm1tN1";
a[51930] = "1svndary"; b[51930] = "O1sond1riX1";
a[51931] = "1syabany"; b[51931] = "O1siy1b1niX1";
a[51932] = "1syasngy"; b[51932] = "O1siy1sangiX1";
a[51933] = "1systany"; b[51933] = "O1sist1niX1";
a[51934] = "1talanta"; b[51934] = "O1tl1nt1Nu";

a[51935] = "1tlantay"; b[51935] = "O1tl1nt1iXu";
a[51936] = "1tlantyk"; b[51936] = "O1tl1ntikNu";
a[51937] = "1tlantys"; b[51937] = "O1tl1ntisNu";
a[51938] = "1trvpyny"; b[51938] = "O1tropiniX1";
a[51939] = "1twbaryy"; b[51939] = "O1tawb1riiX1";
a[51940] = "1twbazyy"; b[51940] = "O1tawb1ziiX1";
a[51941] = "1twcrxan"; b[51941] = "O1tawcarx1nN1";
a[51942] = "1twfwany"; b[51942] = "O1tawfew1niA0";
a[51943] = "1twgyrhy"; b[51943] = "O1tawgireiX1";
a[51944] = "1twkafty"; b[51944] = "O1tawk1fiX1";
a[51945] = "1twkaryy"; b[51945] = "O1tawk1riiX1";
a[51946] = "1twnwany"; b[51946] = "O1tawnew1niN1";
a[51947] = "1twxanhy"; b[51947] = "O1tawx1neiX1";
a[51948] = "1vangard"; b[51948] = "O1v1ng1rdN1";
a[51949] = "1vansyan"; b[51949] = "O1v1n1nsiy1nNo";
a[51950] = "1vanvysy"; b[51950] = "O1v1nevisiN1";
a[51951] = "1vazxvan"; b[51951] = "O1v1zx1nA0";
a[51952] = "1vrdgahy"; b[51952] = "O1vardg1hiX1";
a[51953] = "1vyynyvny"; b[51953] = "O1viniyoniXu";
a[51954] = "1vyxtgyy"; b[51954] = "O1vixtegiX1";
a[51955] = "1w^aldan"; b[51955] = "O1wq1ld1rdN1";
a[51956] = "1wamydny"; b[51956] = "O1w1lmidaniN1";
a[51957] = "1wkargry"; b[51957] = "O1wek1rgariX1";
a[51958] = "1wkarsaz"; b[51958] = "O1wk1rs1zA0";
a[51959] = "1wkarydn"; b[51959] = "O1wek1ridanVv";
a[51960] = "1wnasazy"; b[51960] = "O1wn1s1ziN1";
a[51961] = "1wpzbawy"; b[51961] = "O1wpazb1wiN1";
a[51962] = "1wpzaxanh"; b[51962] = "O1wpazx1neN1";
a[51963] = "1wvbgryy"; b[51963] = "O1wobgariiX1";
a[51964] = "1wvradhy"; b[51964] = "O1wur1deiXu";
a[51965] = "1xalsvzy"; b[51965] = "O1x1lsoziX1";
a[51966] = "1xralnhr"; b[51966] = "O1xeronnahrN1";
a[51967] = "1xreaqbt"; b[51967] = "O1xar1qebatiN1";
a[51968] = "1xvndyan"; b[51968] = "O1xundi1nNo";
a[51969] = "1xwyjany"; b[51969] = "O1xwi1jiniX1";
a[51970] = "1yranwym"; b[51970] = "O1yr1ngwimN1";
a[51971] = "1yrvgram"; b[51971] = "O1y1r1ngwimN1";
a[51972] = "1yrvmtry"; b[51972] = "O1yrometriX1";
a[51973] = "1yznhavr"; b[51973] = "O1yzonh1verNa";
a[51974] = "1zadbtxy"; b[51974] = "O1z1dbaxtiXo";
a[51975] = "1zaddarv"; b[51975] = "O1z1dd1roN1";
a[51976] = "1zaddary"; b[51976] = "O1z1dd1riX1";
a[51977] = "1zaddrxt"; b[51977] = "O1z1dderaxtiN1";
a[51978] = "1zadfky"; b[51978] = "O1z1dfekriN1";
a[51979] = "1zadg@ar"; b[51979] = "O1z1dgoz1rA0";
a[51980] = "1zadgany"; b[51980] = "O1z1deg1niXu";
a[51981] = "1zadkary"; b[51981] = "O1z1dk1riX1";
a[51982] = "1zadmahy"; b[51982] = "O1z1dm1hiN1";
a[51983] = "1zadmjdy"; b[51983] = "O1z1dmajdiXo";
a[51984] = "1zadmnwy"; b[51984] = "O1z1dmanewiN1";
a[51985] = "1zadmrdy"; b[51985] = "O1z1dmardiN1";
a[51986] = "1zadsazy"; b[51986] = "O1z1ds1ziN1";
a[51987] = "1zadwhry"; b[51987] = "O1z1dwahriXu";
a[51988] = "1zardvst"; b[51988] = "O1z1rdostA0";
a[51989] = "1zargryy"; b[51989] = "O1z1rgariiX1";
a[51990] = "1zmayndh"; b[51990] = "O1zm1yandeA0";
a[51991] = "1zmaywgr"; b[51991] = "O1zm1yewgarN1";
a[51992] = "1zmndanh"; b[51992] = "O1zm1d1neAp";
a[51993] = "1zmvngry"; b[51993] = "O1z1mongariX1";
a[51994] = "@by&yany"; b[51994] = "OzabihiiNiXo";

a[51995] = "@valfvnm"; b[51995] = "OzolfonunA0";
a[51996] = "@valfqar"; b[51996] = "Ozolfaq1rA0";
a[51997] = "@valjlal"; b[51997] = "Ozoljal11A0";
a[51998] = "@valjna&"; b[51998] = "Ozoljen1hA0";
a[51999] = "@y\$la&yt"; b[51999] = "Ozisal1hiyyatA0";
a[52000] = "@y&sabby"; b[52000] = "Ozihes1biiX1";
a[52001] = "@asazy"; b[52001] = "Oqaz1s1ziiX1";
a[52002] = "armahyy"; b[52002] = "Oq1rm1hiiX1";
a[52003] = "arnvrdy"; b[52003] = "Oq1rnnavardiN1";
a[52004] = "arnwyny"; b[52004] = "Oq1rnwewiniN1";
a[52005] = "arv^vry"; b[52005] = "Oq1roqoriX1";
a[52006] = "arwnasy"; b[52006] = "Oq1rwen1siN1";
a[52007] = "aryqvny"; b[52007] = "Oq1riqoniX1";
a[52008] = "azsanan"; b[52008] = "Oq1zsin1nN1";
a[52009] = "bar1lvd"; b[52009] = "Oqob1r1ludA0";
a[52010] = "barrvby"; b[52010] = "Oqob1rrubiN1";
a[52011] = "afaryany"; b[52011] = "Oqaff1riy1niXo";
a[52012] = "afvryany"; b[52012] = "Oqafuriy1niXo";
a[52013] = "alam& sny"; b[52013] = "Oqol1mhasaniXa";
a[52014] = "alam&syn"; b[52014] = "Oqol1mhoseynNa";
a[52015] = "alamelyy"; b[52015] = "Oqol1maliiXa";
a[52016] = "lampvry"; b[52016] = "Oqol1mpuriXo";
a[52017] = "lamr#ay"; b[52017] = "Oqol1mrez1iXa";
a[52018] = "lamrsvl"; b[52018] = "Oqol1mrasulNa";
a[52019] = "ltanydn"; b[52019] = "Oqalt1nidanVV";
a[52020] = "l~anydh"; b[52020] = "Oqalt1nideA0";
a[52021] = "l~namhy"; b[52021] = "Oqalatin1meiX1";
a[52022] = "mgsaryy"; b[52022] = "Oqamgos1riiX1";
a[52023] = "mgynanh"; b[52023] = "Oqamgin1neAp";
a[52024] = "mxvaryy"; b[52024] = "Oqamx1riiX1";
a[52025] = "nabxwyy"; b[52025] = "Oqen1baxwiiX1";
a[52026] = "r#mddy"; b[52026] = "OqarazmandiiX1";
a[52027] = "r^rknan"; b[52027] = "Oqorqorkon1nAp";
a[52028] = "rbalgry"; b[52028] = "Oqarb1lgariN1";
a[52029] = "rbgrasty"; b[52029] = "Oqarbger1iN1";
a[52030] = "rjstany"; b[52030] = "Oqarjest1niXu";
a[52031] = "rral&km"; b[52031] = "OqorarolhekamN1";
a[52032] = "asalxanh"; b[52032] = "Oqass1lx1neN1";
a[52033] = "vrba^hy"; b[52033] = "Oqorb1qeiX1";
a[52034] = "ybtgryy"; b[52034] = "OqeybatgariiX1";
a[52035] = "yr#rvry"; b[52035] = "OqeyrezaruriA0";
a[52036] = "yr\$dayy"; b[52036] = "Oqeyresed1yiA0";
a[52037] = "yr\$nety"; b[52037] = "OqeyresanatiA0";
a[52038] = "yr&#vry"; b[52038] = "OqeyrehuzuriA0";
a[52039] = "yr&qvqy"; b[52039] = "OqeyrehuquqiA0";
a[52040] = "yr&qyqy"; b[52040] = "OqeyrehaqiqiA0";
a[52041] = "yr-ntyk"; b[52041] = "Oqeyre7enetikA0";
a[52042] = "yr1mary"; b[52042] = "Oqeyre1m1riA0";
a[52043] = "yr1wkar"; b[52043] = "Oqeyre1wk1rA0";
a[52044] = "yr1yyiny"; b[52044] = "Oqeyre1yiniA0";
a[52045] = "yra\$lvly"; b[52045] = "OqeyreosuliA0";
a[52046] = "yradary"; b[52046] = "Oqeyreed1riA0";
a[52047] = "yranbya"; b[52047] = "Oqeyreanbiy1A0";
a[52048] = "yrrarady"; b[52048] = "Oqeyreer1diA0";
a[52049] = "yrrartwy"; b[52049] = "OqeyreartewiN1";
a[52050] = "yrrazwy"; b[52050] = "OqeyrearezewiA0";
a[52051] = "yrrasasy"; b[52051] = "Oqeyreas1siA0";
a[52052] = "yrawbae"; b[52052] = "Oqeyreewb1A0";
a[52053] = "yraymny"; b[52053] = "OqeyreimeniA0";
a[52054] = "yrbhynh"; b[52054] = "OqeyrebehineA0";
a[52055] = "yrdaymy"; b[52055] = "Oqeyred1yemiA0";
a[52056] = "yrdfaey"; b[52056] = "Oqeyredef1iA0";
a[52057] = "yrdvltv"; b[52057] = "OqeyredolatiA0";
a[52058] = "yrfarsy"; b[52058] = "Oqeyref1rsiA0";
a[52059] = "yrfysyly"; b[52059] = "OqeyrefosiliA0";
a[52060] = "yrhms~&"; b[52060] = "OqeyrehamsathA0";
a[52061] = "yrhvazy"; b[52061] = "Oqeyrehav1ziA0";
a[52062] = "yrtjna&y"; b[52062] = "Oqeyrejen1hiA0";
a[52063] = "yrtjnaay"; b[52063] = "Oqeyrejen1iA0";
a[52064] = "yrtjvhry"; b[52064] = "OqeyrejohariA0";
a[52065] = "yrtjzaay"; b[52065] = "Oqeyrejaz1iA0";
a[52066] = "yrklywh"; b[52066] = "OqeyrekeliweA0";
a[52067] = "yrktaby"; b[52067] = "Oqeyreket1biA0";
a[52068] = "yrkyfry"; b[52068] = "OqeyrekeyfariA0";
a[52069] = "yrm&\$vr"; b[52069] = "OqeyremahsorA0";
a[52070] = "yrm&jvr"; b[52070] = "OqeyremahjorA0";
a[52071] = "yrm&rvm"; b[52071] = "OqeyremahromA0";
a[52072] = "yrm&svs"; b[52072] = "OqeyremahsusA0";
a[52073] = "yrm&tml"; b[52073] = "OqeyremohtamelA0";
a[52074] = "yrm@hby"; b[52074] = "OqeyremazhabiA0";
a[52075] = "yrm@kvr"; b[52075] = "OqeyremazkorA0";
a[52076] = "yrmbtla"; b[52076] = "Oqeyremobtal1A0";
a[52077] = "yrme\$vm"; b[52077] = "OqeyremasumA0";
a[52078] = "yrmehvd"; b[52078] = "OqeyremahudA0";
a[52079] = "yrmemvl"; b[52079] = "OqeyremamulA0";
a[52080] = "yrmenvy"; b[52080] = "OqeyremanaviA0";
a[52081] = "yrmeqvl"; b[52081] = "OqeyremaqulA0";
a[52082] = "yrmqtbl"; b[52082] = "OqeyremotabarA0";
a[52083] = "yrmqtd"; b[52083] = "OqeyremotaqedA0";
a[52084] = "yrmfhvm"; b[52084] = "OqeyremafhomA0";
a[52085] = "yrmnasb"; b[52085] = "Oqeyremon1sebaA0";
a[52086] = "yrmnqvl"; b[52086] = "OqeyremanqulA0";
a[52087] = "yrmnsjm"; b[52087] = "OqeyremonsajemA0";
a[52088] = "yrmn~qy"; b[52088] = "OqeyremanteqiA0";
a[52089] = "yrmqtbl"; b[52089] = "OqeyremaqbolA0";
a[52090] = "yrmrdmy"; b[52090] = "OqeyremardomiA0";
a[52091] = "yrmrsvm"; b[52091] = "OqeyremaresomA0";
a[52092] = "yrmrtb~"; b[52092] = "OqeyremortabetA0";
a[52093] = "yrmr~vb"; b[52093] = "OqeyremartobA0";
a[52094] = "yrmssvl"; b[52094] = "OqeyreemasulA0";
a[52095] = "yrmstdl"; b[52095] = "OqeyremostadalA0";
a[52096] = "yrmstmr"; b[52096] = "OqeyremostamerA0";
a[52097] = "yrmstnd"; b[52097] = "OqeyremostanadA0";
a[52098] = "yrmstql"; b[52098] = "OqeyremostaqelA0";

a[52099] = "^\yrm\sy&y"; b[52099] = "OqeyremasihiA0";
a[52100] = "^\yrm\tdyn"; b[52100] = "OqeyremotedayyenA0";
a[52101] = "^\yrm\tehd"; b[52101] = "OqeyremotaahhedA0";
a[52102] = "^\yrm\tdn"; b[52102] = "OqeyremotemaddenA0";
a[52103] = "^\yrm\tx\$\$"; b[52103] = "OqeyremotexasA0";
a[52104] = "^\yrm\vazy"; b[52104] = "OqeyremovlziA0";
a[52105] = "^\yrm\wabh"; b[52105] = "OqeyremowlbeA0";
a[52106] = "^\yrm\wkvk"; b[52106] = "OqeyremawkokA0";
a[52107] = "^\yrm\wvml"; b[52107] = "OqeyremawmulA0";
a[52108] = "^\yrm\wrve"; b[52108] = "OqeyremawruA0";
a[52109] = "^\yrm\wrv~"; b[52109] = "OqeyremawrotA0";
a[52110] = "^\yrm\!amy"; b[52110] = "Oqeyrenez1miN1";
a[52111] = "^\yrm\fv@y"; b[52111] = "OqeyrenofoziA0";
a[52112] = "^\yrm\hady"; b[52112] = "OqeyrenahlidiA0";
a[52113] = "^\yrm\pzwy"; b[52113] = "OqeyrepezewkiA0";
a[52114] = "^\yrm\q#ayy"; b[52114] = "OqeyreqazlyiA0";
a[52115] = "^\yrm\qrlny"; b[52115] = "OqeyreqorlniA0";
a[52116] = "^\yrm\rv^ny"; b[52116] = "OqeyreroqaniA0";
a[52117] = "^\yrm\rsa&ly"; b[52117] = "OqeyreslheliiA0";
a[52118] = "^\yrm\rsvary"; b[52118] = "OqeyresavlriA0";
a[52119] = "^\yrm\rsyasy"; b[52119] = "Oqeyresiy1siA0";
a[65323] = "nykvkaranh"; b[65323] = "Onikok1r1neAp";
a[65324] = "nylvfryany"; b[65324] = "Onilofary1niX1";
a[65325] = "nyrngstany"; b[65325] = "Oneyrangest1niX1";
a[65326] = "nyrvmdandh"; b[65326] = "Oniromand1neAp";
a[65327] = "nytrasyvny"; b[65327] = "Onitrl1sionyX1";
a[65328] = "nyvanglndy"; b[65328] = "OniyuengelandiXu";
a[65329] = "nyvmkzykvy"; b[65329] = "OniyomekzikoiXu";
a[65330] = "padnvtvny"; b[65330] = "Op1dnutroniX1";
a[65331] = "padprvtvny"; b[65331] = "Op1dporotoniX1";
a[65332] = "padrmyany"; b[65332] = "Op1darmiy1niiX1";
a[65333] = "palaywgahy"; b[65333] = "Op1llyewg1hiA0";
a[65334] = "pallvgrafy"; b[65334] = "Op1leoger1fiN1";
a[65335] = "pallvlytyk"; b[65335] = "Op1leolitikN1";
a[65336] = "pallvzvvyk"; b[65336] = "Op1leozoikN1";
a[65337] = "palvgrafyy"; b[65337] = "Op1leoger1fiiX1";
a[65338] = "palvlytyky"; b[65338] = "Op1leolitikiX1";
a[65339] = "palyvgrafy"; b[65339] = "Op1leoger1fiN1";
a[65340] = "palyvlytyk"; b[65340] = "Op1leolitikN1";
a[65341] = "palyvzvvyk"; b[65341] = "Op1leozoikN1";
a[65342] = "panasvnyky"; b[65342] = "Op1n1sonikiX1";
a[65343] = "pankratynny"; b[65343] = "Op1nker1tiniX1";
a[65344] = "pantvgrafy"; b[65344] = "Op1ntoger1fiX1";
a[65345] = "paralmpyky"; b[65345] = "Op1rolampikiX1";
a[65346] = "paranvvydy"; b[65346] = "Op1r1noidiX1";
a[65347] = "paranvyayy"; b[65347] = "Op1r1noylyiN1";
a[65348] = "parsan-ady"; b[65348] = "Op1rs1ne71diXo";
a[65349] = "parsayanhy"; b[65349] = "Op1rs1y1neiX1";
a[65350] = "parvpayany"; b[65350] = "Op1rop1y1niX1";
a[65351] = "pasdarxanh"; b[65351] = "Op1sd1rx1neN1";
a[65352] = "pasyflvryn"; b[65352] = "Op1siflorinN1";
a[65353] = "paydarsazy"; b[65353] = "Op1yd1rs1ziN1";
a[65354] = "pdr1mrzydh"; b[65354] = "Opedar1morzideA0";
a[65355] = "pdrsalaryy"; b[65355] = "Opedars1lriiX1";
a[65356] = "pdrsvxtgyy"; b[65356] = "OpedarsoxtegiX1";
a[65357] = "pdrxvandhy"; b[65357] = "Opedarx1ndeiX1";
a[65358] = "phlvanyany"; b[65358] = "Opahlev1niy1niXo";
a[65359] = "plantarymy"; b[65359] = "Opel1net1riyomiX1";
a[65360] = "plasmvdyvm"; b[65360] = "Opel1smodyomN1";
a[65361] = "plasmvlyzy"; b[65361] = "Opel1smoliziX1";
a[65362] = "platynvvyd"; b[65362] = "Opel1tinoidN1";
a[65363] = "pllystvsny"; b[65363] = "OpeleistoseniX1";
a[65364] = "plvralysmy"; b[65364] = "Opolor1lismiX1";
a[65365] = "plvralysty"; b[65365] = "Opolor1listiA0";
a[65366] = "plvralyzmy"; b[65366] = "Opolor1lizmiX1";
a[65367] = "plvtvnyvmy"; b[65367] = "OpolotoniomyX1";
a[65368] = "pnhankaryy"; b[65368] = "Openh1nk1riiX1";
a[65369] = "pnsylvanya"; b[65369] = "Openselv1niy1Nu";
a[65370] = "pnsylvany"; b[65370] = "Opensilv1niyanN1";
a[65371] = "prbarsazyy"; b[65371] = "Oporb1rs1ziiX1";
a[65372] = "prdaxtkary"; b[65372] = "Opard1xtk1riX1";
a[65373] = "prhyzgaryy"; b[65373] = "Oparhizg1riiX1";
a[65374] = "prhyzkaryy"; b[65374] = "Oparhizk1riiX1";
a[65375] = "prkambryny"; b[65375] = "Operek1mberiyaniX1";
a[65376] = "prtvaufwany"; b[65376] = "Opartoafw1niN1";
a[65377] = "prtvdramany"; b[65377] = "Opartovdarm1niN1";
a[65378] = "prtvgaryy"; b[65378] = "Opartoneg1riiX1";
a[65379] = "prtvwasy"; b[65379] = "Opartowen1siiX1";
a[65380] = "prv-strvny"; b[65380] = "Opero7esteroniX1";
a[65381] = "prvarbndyy"; b[65381] = "Oparv1rbandiiX1";
a[65382] = "prvltaryay"; b[65382] = "Oporolt1ry1iX1";
a[65383] = "prvpagandy"; b[65383] = "Oporop1g1ndiX1";
a[65384] = "prvstrvyka"; b[65384] = "Oporostorik1N1";
a[65385] = "prvtvplasm"; b[65385] = "Oportopel1smN1";
a[65386] = "prvtvplast"; b[65386] = "Oporotopel1stN1";
a[65387] = "prxawgranh"; b[65387] = "Oparx1wgar1neAp";

a[65388] = "psamdrnysm"; b[65388] = "a[65427] = "pywnhadaty"; b[65427] = "Opas1modernismN1";
a[65389] = "psamdrnyst"; b[65389] = "a[65428] = "q#aqvrky"; b[65428] = "Opas1modernistA0";
a[65390] = "psrxvandgy"; b[65390] = "a[65429] = "q\$rfyrvzhy"; b[65429] = "OqasrefiruzeiXu";
"OpesarxIndegiN1"; a[65391] = "psrxvandhy"; b[65391] = "a[65430] = "qabvsnamhy"; b[65430] = "OpesarxIndeiX1";
a[65392] = "ptansyvmtr"; b[65392] = "a[65431] = "qanvng@ary"; b[65431] = "OpetInsiyometrN1";
a[65393] = "ptrvdaktyl"; b[65393] = "Opeterod1ktilN1"; a[65432] = "qanvngraay"; b[65432] = "Oq1nunger1iN1";
a[65394] = "ptrvgrafyy"; b[65394] = "Opetroger1fiiX1"; a[65433] = "qanvnggray"; b[65433] = "Oq1nunger1yiN1";
a[65395] = "pvlkbalany"; b[65395] = "Opolakb111niX1"; a[65434] = "qanvnmndyy"; b[65434] = "Oq1nonmandiiX1";
a[65396] = "pvlkdarany"; b[65396] = "Opolakd1r1niX1"; a[65435] = "qdrnawnasy"; b[65435] = "Oqadrn1wen1siN1";
a[65397] = "pvrasdallh"; b[65397] = "Oporasadoll1hNo"; a[65436] = "qdrnwnasy"; b[65436] = "Oqadrnawn1siiX1";
a[65398] = "pvrvalvvy"; b[65398] = "Opurc1l1uiXo"; a[65437] = "qdrtmndanh"; b[65437] = "Opork1wiyl1niXo";
a[65399] = "pvrdarayy"; b[65399] = "Opor1r1yiiXo"; a[65438] = "qldrnnamhy"; b[65438] = "OpormasomiiXo";
a[65400] = "pvrfrhadyy"; b[65400] = "Oporfarh1diiXo"; a[65439] = "qlyaxvahy"; b[65439] = "Oqaly1x1hiiX1";
a[65401] = "pvrkawyany"; b[65401] = "Oqalandarn1meiX1"; a[65440] = "qmrmlmky"; b[65440] = "OqamarolmolokiXa";
"Opork1wiyl1niXo"; a[65402] = "pvrme\$vmyy"; b[65402] = "Oqalandarn1meiX1"; a[65441] = "qmrdrqrby"; b[65441] = "OqamaradarqabiX1";
"OpormasomiiXo"; a[65403] = "pvrnvgrafy"; b[65403] = "Opornoger1fiN1"; a[65442] = "qnsvlxanhy"; b[65442] = "Oqansolx1neiX1";
a[65404] = "pvrriyany"; b[65404] = "Opurr1jiy1niXo"; a[65443] = "qraatxanhy"; b[65443] = "Oqar1atx1neiX1";
a[65405] = "pvrSanta-y"; b[65405] = "Opors1nt17iX1"; a[65444] = "qraqvynlv"; b[65444] = "Oqar1goyonluiX1";
a[65406] = "pvrwyrzazy"; b[65406] = "Oporwir1ziiXo"; a[65445] = "qrarvmdary"; b[65445] = "Oqar1romad1riX1";
a[65407] = "pvrtrfrvwy"; b[65407] = "OposterforowiN1"; a[65446] = "qrqyzstany"; b[65446] = "Oqerqizest1niXu";
"OposterforowiN1"; a[65408] = "pvstxvaran"; b[65408] = "Opostx1r1nN1"; a[65447] = "qrv^rbylhy"; b[65447] = "OqeroqaribileiX1";
a[65409] = "pvyanmayy"; b[65409] = "Opoy1nem1yiiX1"; a[65448] = "qvnsvlgryy"; b[65448] = "OqonsolgariiX1";
"Opoy1nem1yiiX1"; a[65410] = "pvzytyvysm"; b[65410] = "OpozitivizmN1"; a[65449] = "qvnsvlxanh"; b[65449] = "Oqonsolx1neN1";
a[65411] = "py^mbrany"; b[65411] = "Opeyqambari1niXo"; a[65450] = "qy\$grgrayy"; b[65450] = "OqeySarger1yiiX1";
"Opeyqambari1niXo"; a[65412] = "py^mbrzadh"; b[65412] = "Opeyqambar1deA0"; a[65451] = "q~argrftgy"; b[65451] = "Oqat1rgereftegiN1";
"Opeyqambar1deA0"; a[65413] = "pyambrgvnh"; b[65413] = "Opay1mbarguneA0"; a[65452] = "r#aytmndyy"; b[65452] = "Orez1yatmandiiX1";
"Opay1mbarguneA0"; a[65414] = "pyazkarany"; b[65414] = "Opay1mbarguneA0"; a[65453] = "radykalysm"; b[65453] = "Or1dik1lismN1";
"Opay1mbarguneA0"; a[65415] = "pydzadany"; b[65415] = "Opay1mbarguneA0"; a[65454] = "radyv1ktyv"; b[65454] = "Or1diyoy1ktivN1";
"Opay1mbarguneA0"; a[65416] = "pygrdknndh"; b[65416] = "Opay1mbarguneA0"; a[65455] = "radyvaktv"; b[65455] = "Or1diyoyaktivN1";
"Opay1mbarguneA0"; a[65417] = "pykrtrawyy"; b[65417] = "Opay1mbarguneA0"; a[65456] = "radyvgrafy"; b[65456] = "Or1diyoyger1fiN1";
"Opay1mbarguneA0"; a[65418] = "pymankaryy"; b[65418] = "Opay1mbarguneA0"; a[65457] = "radyvgramy"; b[65457] = "Or1diyoyger1miX1";
"Opay1mbarguneA0"; a[65419] = "pyramvnyan"; b[65419] = "Opay1mbarguneA0"; a[65458] = "radyvl-yst"; b[65458] = "Or1diyoyo7istN1";
"Opay1mbarguneA0"; a[65420] = "pyrapzwkyy"; b[65420] = "Opay1mbarguneA0"; a[65459] = "radyvlv-yy"; b[65459] = "Or1diyolo7iiX1";
"Opay1mbarguneA0"; a[65421] = "pyrmradyan"; b[65421] = "Opay1mbarguneA0"; a[65460] = "radyvpyamy"; b[65460] = "Or1diyoyopay1miX1";
"Opay1mbarguneA0"; a[65422] = "pyrvz1bady"; b[65422] = "Opay1mbarguneA0"; a[65461] = "radyvsazy"; b[65461] = "Or1diyoy1ziiX1";
"Opay1mbarguneA0"; a[65423] = "pyrymydyny"; b[65423] = "Opay1mbarguneA0"; a[65462] = "radyvskppy"; b[65462] = "Or1diyoyoskpiiX1";
"Opay1mbarguneA0"; a[65424] = "pytkantrvp"; b[65424] = "Opay1mbarguneA0"; a[65463] = "radyvtlspk"; b[65463] = "Or1diyoyoteleskopN1";
"Opay1mbarguneA0"; a[65425] = "pytzafrvwy"; b[65425] = "Opay1mbarguneA0"; a[65464] = "radyvtrapy"; b[65464] = "Or1diyoyotor1piN1";
"Opay1mbarguneA0"; a[65426] = "pyvndnamhy"; b[65426] = "Opay1mbarguneA0";
"Opay1mbarguneA0";

a[65465] = "radyvwymy"; b[65465] = a[65506] = "santylytry"; b[65506] = "OsIntilitriX1";
 "Or1diyowimiiX1"; a[65507] = "sanytgrady"; b[65507] = "OsIntiger1diX1";
 a[65466] = "rahdarxanh"; b[65466] = "Or1hd1rx1neN1"; a[65508] = "sarkvdyay"; b[65508] = "Os1rkodin1iX1";
 a[65467] = "rahpymaay"; b[65467] = a[65509] = "saypadyzly"; b[65509] = "Os1yp1dizeliX1";
 "Or1hpeym1iiX1"; a[65510] = "sazmandhy"; b[65510] =
 a[65468] = "rahpymayy"; b[65468] = "Os1zm1ndehiiX1";
 "Or1hpeym1yiiX1"; a[65511] = "sazwkaranh"; b[65511] =
 a[65469] = "rasta&syny"; b[65469] = "Os1zewk1r1neAp";
 "Or1st1hoseyniAp"; a[65512] = "seadtmdny"; b[65512] =
 a[65470] = "rasvsanany"; b[65470] = "Or1sos1n1niX1"; "Osa1datmandiiX1";
 a[65471] = "rbdvwambry"; b[65471] = a[65513] = "sedalsl-nh"; b[65513] = "OsadossaltaneNa";
 "Orobdow1mriX1"; a[65514] = "sfartxanh"; b[65514] = "Osef1ratx1neiX1";
 a[65472] = "rftargray"; b[65472] = "Oraft1rger1yiN1"; a[65515] = "sfarwnamhy"; b[65515] =
 a[65473] = "rftarwnasy"; b[65473] = "Oraft1rwen1siN1"; "Osef1rewn1meiX1";
 a[65474] = "rkvrwdkny"; b[65474] = a[65516] = "sfydbalany"; b[65516] = "Osefidb1l1niX1";
 "Orekordwek1niiX1"; a[65517] = "skvlarysmy"; b[65517] = "Osekol1rismiX1";
 a[65475] = "rngbaxtgyy"; b[65475] = a[65518] = "skvlaryzmy"; b[65518] = "Osekol1rizmiX1";
 "Orangb1xtegiX1"; a[65519] = "slymanpvry"; b[65519] =
 a[65476] = "rv##alwhda"; b[65476] = "Osoley1npuriXo";
 "OrowzatowwohadN1"; a[65520] = "smpatyzy"; b[65520] =
 a[65477] = "rv#talwhda"; b[65477] = "Osamp1tiz1niX1";
 "Orovzatowwohad1N1"; a[65521] = "smsarzhady"; b[65521] =
 a[65478] = "rvdbarany"; b[65478] = "Orudb1r1niiXo"; "Osems1rz1deiXo";
 a[65479] = "rvdrbaysty"; b[65479] = "Orudarb1yestiN1"; a[65522] = "snaryvny"; b[65522] =
 a[65480] = "rvstanwyny"; b[65480] = "Orost1newiniN1"; "Osen1riyonevisN1";
 a[65481] = "rvwnfkranh"; b[65481] = a[65523] = "snaryvsazy"; b[65523] =
 "Orovwanfekr1neAp"; "Osen1riyos1ziN1";
 a[65482] = "rvwnfkrnma"; b[65482] = a[65524] = "sndykalysm"; b[65524] =
 "Orovwanfekrmam1A0"; "Osandik1lismN1";
 a[65483] = "rvwnzadany"; b[65483] = a[65525] = "snkrvrvny"; b[65525] = "OsankrotroniX1";
 "Orowanz1d1niX1"; a[65526] = "sphsaryy"; b[65526] = "Osepahs1l1riiX1";
 a[65484] = "rvygrdany"; b[65484] = "Oruygard1niiX1"; a[65527] = "spntamynvy"; b[65527] =
 a[65485] = "rvzyzynysm"; b[65485] = "Osepant1minoiX1";
 "OrevizyonismN1"; a[65528] = "srafkndgyy"; b[65528] =
 a[65486] = "rvzyvny"; b[65486] = "OrevizyonistA0"; "OsarafkandegiiX1";
 a[65487] = "rvzwxarany"; b[65487] = a[65529] = "srafrazanh"; b[65529] = "Osarifr1z1neAp";
 "Orozwek1r1niX1"; a[65530] = "srbazgyry"; b[65530] = "Osarb1zgiriiX1";
 a[65488] = "rwydaldyny"; b[65488] = a[65531] = "srbazxanh"; b[65531] = "Osarb1zx1neiX1";
 "OrawidoddiniX1"; a[65532] = "srdftardary"; b[65532] = "Osardaftard1riX1";
 a[65489] = "rxtardarxanh"; b[65489] = "Oraxtd1rx1neN1"; a[65533] = "srdstynwyn"; b[65533] =
 a[65490] = "rxtwvrxanh"; b[65490] = "OsardsirnewinA0";
 "Oraxtwurx1neN1"; a[65534] = "srfmandhy"; b[65534] =
 a[65491] = "rxtwvxanh"; b[65491] = "Osarfarm1ndehiN1";
 "Oraxtvox1neiX1"; a[65535] = "srgrvhbany"; b[65535] =
 a[65492] = "rxtwvyanh"; b[65492] = "Osargorohb1niN1";
 "Oraxtwuyx1neN1"; a[65536] = "srknsvlgyy"; b[65536] =
 a[65493] = "rybvflavyn"; b[65493] = "Oribofel1vinN1"; "Osarkonsolgarin1";
 a[65494] = "ryxtgrzadh"; b[65494] = "Orixtegarz1deNo"; a[65537] = "srmadrmany"; b[65537] =
 a[65495] = "sakarvmyst"; b[65495] = "Os1k1romistN1"; "Osarm1darm1niN1";
 a[65496] = "saladxvryy"; b[65496] = "Os1l1dxoriiX1"; a[65538] = "srmajra&yy"; b[65538] =
 a[65497] = "salbvtamv1"; b[65497] = "Os1lbut1molN1"; "Osarm1jarr1hiiX1";
 a[65498] = "salmvnlvzy"; b[65498] = a[65539] = "srmavrdgy"; b[65539] =
 "Os1lmonelloziX1"; "Osarm1xordegiN1";
 a[65499] = "salxvrdgyy"; b[65499] = "Os1lxordegiX1"; a[65540] = "srmhmandar"; b[65540] =
 a[65500] = "salysylaty"; b[65500] = "Os1lil1itiX1"; "Osarmehm1nd1rN1";
 a[65501] = "sanatvrymy"; b[65501] = a[65541] = "srrsydnamh"; b[65541] =
 "Os1n1toryomiX1"; "Osarresidn1meN1";
 a[65502] = "santralysm"; b[65502] = "Os1n1lismN1"; a[65542] = "srdvdxvany"; b[65542] = "Osurudx1niiX1";
 a[65503] = "santrvzvm"; b[65503] = "Os1ntrozomiX1"; a[65543] = "srmkanysm"; b[65543] =
 a[65504] = "santryfy-y"; b[65504] = "Os1nterifo7iX1"; "Oservomek1nismN1";
 a[65505] = "sanytgrady"; b[65505] = "OsIntiger1diX1";

a[65544] = "stbrnvkany"; b[65544] = a[65586] = "t\$vyrngary"; b[65586] = "Otasvirmeg1riX1";
 "Osetabrnok1niX1"; a[65587] = "t\$vyprdz"; b[65587] = "Otasvirpard1zN1";
 a[65545] = "svad1mvzyy"; b[65545] = a[65588] = "t\$vyrsazy"; b[65588] = "Otasvirs1ziiX1";
 "Osav1d1muziiX1"; a[65589] = "t&\$yldary"; b[65589] = "Otahsild1riiX1";
 a[65546] = "svastfady"; b[65546] = "Osoestef1deiX1"; a[65590] = "t&jrgrayy"; b[65590] =
 a[65547] = "svgndnamhy"; b[65547] = "Otahajjorger1yiiX1";
 "Osogandn1meiX1"; a[65591] = "t&lylgranh"; b[65591] = "Otahlilgar1neAp";
 a[65548] = "svkmysyvny"; b[65548] = a[65592] = "t&vyldary"; b[65592] = "Otahvild1riiX1";
 "OsokomisyoniX1"; a[65593] = "t^yyrdhndh"; b[65593] =
 a[65549] = "svnvgrafyy"; b[65549] = "Osonoger1fiiX1"; "OtaqyirdahandeA0";
 a[65550] = "svnvl-ysty"; b[65550] = "Osonolo7istiX1"; a[65594] = "t^yyryaby"; b[65594] = "Otaqyiry1biiX1";
 a[65551] = "svprastary"; b[65551] = "Osoperest1riN1"; a[65595] = "ta%yrangyz"; b[65595] = "OtasirangizA0";
 a[65552] = "svrralysmy"; b[65552] = "Osorere1limiX1"; a[65596] = "ta%yrg@ary"; b[65596] = "Otasirgoz1riN1";
 a[65553] = "svrralysm"; b[65553] = "Osurere1limN1"; a[65597] = "ta%yrgyry"; b[65597] = "OtasirgiriX1";
 a[65554] = "svrralyst"; b[65554] = "Osurere1listA0"; a[65598] = "ta%yrsazy"; b[65598] = "Otasirs1ziiX1";
 a[65555] = "svsmarmahy"; b[65555] = a[65599] = "tablvnvysy"; b[65599] = "Ot1blonevisiN1";
 "Ososm1rm1hiN1"; a[65600] = "tajykstany"; b[65600] = "Ot1jikest1niA0";
 a[65556] = "svspansyvn"; b[65556] = a[65601] = "taksydrmy"; b[65601] = "Ot1ksidermiX1";
 "Ososp1nsiyonN1"; a[65602] = "talarxanhy"; b[65602] = "Ot1l1rx1neiX1";
 a[65557] = "svsyalysmy"; b[65557] = "Osusiy1lismiX1"; a[65603] = "tarvtbnvry"; b[65603] = "Ot1rotamboriX1";
 a[65558] = "svsyalyst"; b[65558] = "Osusiy1listiA0"; a[65604] = "taryxsazy"; b[65604] = "Ot1rixs1ziiX1";
 a[65559] = "svsyalyzmy"; b[65559] = "Osusiy1lizmiX1"; a[65605] = "tatarstany"; b[65605] = "Ot1t1rest1niXu";
 a[65560] = "svvbrdawty"; b[65560] = a[65606] = "tbarwnasy"; b[65606] =
 "Osuebard1wtiX1"; "Otab1rwen1siiX1";
 a[65561] = "sxavtmndy"; b[65561] = a[65607] = "tbeydkndh"; b[65607] =
 "Osex1vatmandiiX1"; "OtabidkonandeA0";
 a[65562] = "sxtpvstany"; b[65562] = "Osaxtpost1niX1"; a[65608] = "tblvryafth"; b[65608] = "Otabalvory1fteA0";
 a[65563] = "syastg@ary"; b[65563] = a[65609] = "tbxyrknndh"; b[65609] =
 "Osiy1satgoz1riN1"; "OtabxirkonandeN1";
 a[65564] = "syastmdary"; b[65564] = a[65610] = "teddgraayy"; b[65610] =
 "Osiy1satmad1riN1"; "Otaaddodgar1iiX1";
 a[65565] = "syastnamhy"; b[65565] = a[65611] = "teddgrayy"; b[65611] =
 "Osiy1satn1meiX1"; "Otaaddodger1yiiX1";
 a[65566] = "sydalsadat"; b[65566] = a[65612] = "tehdp@yry"; b[65612] =
 "Oseyyedoss1d1tNa"; "OtaahhodpaziriiX1";
 a[65567] = "sygarfrvwy"; b[65567] = "Osig1rforowiX1"; a[65613] = "temyddhndh"; b[65613] =
 a[65568] = "sygarsazy"; b[65568] = "Osig1rs1ziiX1"; "OtamiddahandeN1";
 a[65569] = "syklvtrvny"; b[65569] = "OsiklotroniX1"; a[65614] = "temyrkary"; b[65614] = "Otamirk1riiX1";
 a[65570] = "symankary"; b[65570] = "Osim1nk1riiX1"; a[65615] = "temyrknndh"; b[65615] =
 a[65571] = "synkrvtrvn"; b[65571] = "OsinkrotronN1"; "OtamirkonandeA0";
 a[65572] = "synmadary"; b[65572] = a[65616] = "tezyrwvndh"; b[65616] =
 "Osinem1d1riiX1"; "OtazirwavandeA0";
 a[65573] = "synmaprdaz"; b[65573] = a[65617] = "tfsyrknndh"; b[65617] =
 "Osinam1pard1zN1"; "OtafsirkonandeA0";
 a[65574] = "synmasazy"; b[65574] = a[65618] = "tfsyrgary"; b[65618] = "Otafsirmeg1riN1";
 "Osinam1s1ziiX1"; a[65619] = "tfsyrwnasy"; b[65619] = "Otafsirwen1siN1";
 a[65575] = "synmaskvpy"; b[65575] = a[65620] = "thdyknndh"; b[65620] =
 "Osinem1eskopiX1"; "OtahdidkonandeA0";
 a[65576] = "synsvvydy"; b[65576] = "OsinusoidiX1"; a[65621] = "thdydnamhy"; b[65621] =
 a[65577] = "syralmlvky"; b[65577] = "Otahdidn1meiX1"; "Otahdidn1meiX1";
 "OseyrolmolokiX1"; a[65622] = "thdydwvndh"; b[65622] =
 a[65578] = "syrstyrkhay"; b[65578] = "Osirsirakh1iX1"; "OtahdidwavandeA0";
 a[65579] = "syrsvgrafy"; b[65579] = "Osismoger1fiX1"; a[65623] = "thranparsy"; b[65623] = "Otehr1np1rsiXu";
 a[65580] = "sytvasyvny"; b[65580] = "Ositv1siyoniX1"; a[65624] = "tjartxanhy"; b[65624] = "Otej1ratx1neiX1";
 a[65581] = "sytvplasm"; b[65581] = "Ositope1lsmiX1"; a[65625] = "tjavzgranh"; b[65625] =
 a[65582] = "t!ahrknndh"; b[65582] = "Otaj1vozgar1neA0";
 "Otaj1horkonandeA0"; a[65626] = "tjavz~lbyy"; b[65626] =
 a[65583] = "t\$mymmsazy"; b[65583] = "Otaj1voztalabiiX1"; "Otaj1voztalabiiX1";
 "Otasimms1ziN1"; a[65627] = "tjddgrayy"; b[65627] =
 a[65584] = "t\$vyrbdrar"; b[65584] = "Otasvirbard1riN1"; "Otajaddodger1yiiX1";
 a[65585] = "t\$vyrgdan"; b[65585] = "Otasvirgard1nN1";

a[65628] = "tjddxvahyy"; b[65628] = a[65666] = "trkmancayy"; b[65666] =
 "Otajaddodx1hiiX1"; a[65667] = "trkmnstany"; b[65667] =
 a[65629] = "tjdydp@ryr"; b[65629] = "OtajdidpaziriN1";
 a[65630] = "tk%rgraayy"; b[65630] = "Otokamanest1niA0";
 "Otakassorger1iiX1"; a[65668] = "trmvtrapyy"; b[65668] =
 a[65631] = "tk%rgrayyy"; b[65631] = "Otermotor1piiX1";
 "Otakassorger1yiiX1"; a[65669] = "trmvvyvnyky"; b[65669] =
 a[65632] = "tkafvknndh"; b[65632] = "OtermoyonikiX1";
 "Otak1fokonandeA0"; a[65670] = "trmynatvry"; b[65670] = "Otermin1toriX1";
 a[65633] = "tknkvkrasyy"; b[65633] = "Oteknoker1siiX1"; a[65671] = "trmynvl-yy"; b[65671] =
 a[65634] = "tknvl-ysty"; b[65634] = "Oteknolo7istiX1"; "Oterminolo7iiX1";
 a[65635] = "tknvlv-yky"; b[65635] = "Oteknolo7ikiA0"; a[65672] = "trvmbvsvy"; b[65672] = "OtorombositiX1";
 a[65636] = "tkvandvkar"; b[65636] = a[65673] = "trvtskysmy"; b[65673] = "OteroteskismiX1";
 "Otekv1ndok1rN1"; a[65674] = "trykvbafyy"; b[65674] = "Oterikob1fiiX1";
 a[65637] = "tkyaxtgany"; b[65637] = a[65675] = "trylvbytha"; b[65675] = "Oterilobith1N1";
 "Otakylxtegl1niX1"; a[65676] = "tststvrny"; b[65676] = "OtestosteroniX1";
 a[65638] = "tmawagnma"; b[65638] = a[65677] = "tt%yrangyz"; b[65677] = "OtasirangizA0";
 "Otam1wl1garnam1A0"; a[65678] = "tt%yrg@ary"; b[65678] = "Otasirgoz1riN1";
 a[65639] = "tmawaknndh"; b[65639] = a[65679] = "tt%yrgyryy"; b[65679] = "OtasirgiriX1";
 "Otam1wl1konandeA0"; a[65680] = "tt%yrsazy"; b[65680] = "Otasirs1ziiX1";
 a[65640] = "tmawaxanhy"; b[65640] = a[65681] = "ttatrwasy"; b[65681] = "Ote1trwen1siX1";
 "Otam1wl1neiX1"; a[65682] = "tv&ydxanhy"; b[65682] = "Otohidx1neiX1";
 a[65641] = "tmayzg@ary"; b[65641] = a[65683] = "tvanasazy"; b[65683] = "Otav1n1s1ziiX1";
 "Otam1yozgoz1riN1"; a[65684] = "tvbyxnamhy"; b[65684] =
 a[65642] = "tmazyabyy"; b[65642] = "Otohibx1meiX1";
 "Otam1yozy1biiX1"; a[65685] = "tvlvvydiny"; b[65685] = "OtoloidiniX1";
 a[65643] = "tmayz~lbyy"; b[65643] = a[65686] = "tvlyddarvy"; b[65686] = "Otolidd1roiX1";
 "Otam1yoztalabiiX1"; a[65687] = "tvlydknndh"; b[65687] =
 a[65644] = "tmbrwnasy"; b[65644] = "OtovlidkonandeA0";
 "Otamrwen1siiX1"; a[65688] = "tvmvgrafyy"; b[65688] = "Otomoger1fiiX1";
 a[65645] = "tmrkzdhdh"; b[65645] = a[65689] = "tvpvgrafyy"; b[65689] = "Otopoger1fiiX1";
 "OtamarkozdahandeA0"; a[65690] = "tvtalytary"; b[65690] = "Otolit1riX1";
 a[65646] = "tmrkzgraay"; b[65646] = a[65691] = "tvtvncyany"; b[65691] = "Ototonciy1niXo";
 "Otamarkozger1iN1"; a[65692] = "txmyrkndh"; b[65692] =
 a[65647] = "tmrkzgrayy"; b[65647] = "OtaxmirkonandeA0";
 "Otamarkozger1yiN1"; a[65693] = "tymardary"; b[65693] = "Otim1rd1riiX1";
 a[65648] = "tmrkzdaay"; b[65648] = a[65694] = "tymarknndh"; b[65694] =
 "Otamarkozod1iN1"; a[65695] = "tymarstany"; b[65695] = "Otim1rest1niX1";
 a[65649] = "tmrkzdday"; b[65649] = a[65696] = "tymvrzadhy"; b[65696] =
 "Otamarkozod1yiN1"; a[65697] = "tymardary"; b[65697] = "Otim1rd1riiX1";
 a[65650] = "tmystvklisy"; b[65650] = "OtemistokelsiXa"; "Oteymurz1deiXo";
 a[65651] = "tnabdarany"; b[65651] = "Otan1bd1r1niX1"; a[65698] = "tyrandazy"; b[65698] = "Otirand1ziiX1";
 a[65652] = "tqa#aknndh"; b[65652] = a[65699] = "tyrtxwayyy"; b[65699] = "Otiraxw1yiiX1";
 "Otaqlz1konandeA0"; a[65700] = "tytrasyvny"; b[65700] = "Otitr1siyoniX1";
 a[65653] = "tqa#anamhy"; b[65653] = a[65701] = "tzryqatcy"; b[65701] = "Otaqriq1tciX1";
 "Otaqlz1n1meiX1"; a[65702] = "v&ydalmlky"; b[65702] =
 a[65654] = "tqa#awvndh"; b[65654] = "OvahidolmolkiXa";
 "Otaqlz1wandeA0"; a[65703] = "valrwtayny"; b[65703] =
 a[65655] = "tqdyrgrayy"; b[65655] = "Otaqdirger1yiN1"; "Ov1lerewt1yniXa";
 a[65656] = "tqdyrnamhy"; b[65656] = a[65704] = "valybalyst"; b[65704] = "Ov1lib1listN1";
 "Otaqdirn1meiX1"; a[65705] = "vaqegraayy"; b[65705] = "Ov1qeger1iiX1";
 a[65657] = "tqlydknndh"; b[65657] = a[65706] = "vayznamhy"; b[65706] =
 "OtaqlidkonandeA0"; a[65707] = "vaystadnky"; b[65707] =
 a[65658] = "traktvrcyy"; b[65658] = "Oter1ktorciX1"; "Ov1lrist1danakiAp";
 a[65659] = "traktvrsaz"; b[65659] = "Oter1ktors1zN1"; a[65708] = "vlvntarysm"; b[65708] = "Ovolont1rismN1";
 a[65660] = "transpvrt"; b[65660] = "Oter1nsportiX1"; a[65709] = "vlvntaryst"; b[65709] = "Ovolont1ristA0";
 a[65661] = "tranzystvr"; b[65661] = "Oter1nzistorN1"; a[65710] = "vrdarvrmal"; b[65710] =
 a[65662] = "trazvdary"; b[65662] = "Otar1zod1riiX1"; "Ovard1rvarm1IA0";
 a[65663] = "trazvsazy"; b[65663] = "Otar1zos1ziiX1";
 a[65664] = "trdyd1vryy"; b[65664] = "Otaridid1variiX1";
 a[65665] = "trdydafkny"; b[65665] = "OtarididafkaniN1";

a[65711] = "vstfalyayy"; b[65711] = "Ovestf1liiy1yiA0";	a[65744] = "wvfa-xanhy"; b[65744] =
a[65712] = "vyddvfrvwy"; b[65712] =	"Owof17x1neiX1";
"OvideofuruwiN1";	a[65745] = "wvhrxvahy"; b[65745] =
a[65713] = "vyddvklvby"; b[65713] =	"Owoharx1hariX1";
"OvideokulubiX1";	a[65746] = "wvrangzyy"; b[65746] = "OworangiziiX1";
a[65714] = "vydvfrvwy"; b[65714] =	a[65747] = "wvrayealyy"; b[65747] = "Owor1ye1liiX1";
"OvideoforowiiX1";	a[65748] = "wyrdryayyy"; b[65748] =
a[65715] = "vydyvfrvwy"; b[65715] =	"Owiredary1yiiX1";
"OvideoforowiN1";	a[65749] = "wyrkakaavy"; b[65749] = "Owirklk1uiX1";
a[65716] = "vylcrranyy"; b[65716] = "Ovilcerr1niiX1";	a[65750] = "wyrpzxanhy"; b[65750] =
a[65717] = "vyrastaryy"; b[65717] = "Ovir1st1riiX1";	"Owirepax1neiX1";
a[65718] = "vyskvzythy"; b[65718] = "OviskoziteiX1";	a[65751] = "wyrxvargah"; b[65751] = "Owirx1rg1hN1";
a[65719] = "vzartxanhy"; b[65719] =	a[65752] = "wyrxvargyy"; b[65752] = "Owirx1regiiX1";
"Ovez1ratx1neiX1";	a[65753] = "xakstrnwyn"; b[65753] =
a[65720] = "vzartxarjh"; b[65720] =	"Ox1kestarnewinA0";
"Ovez1ratex1rejeN1";	a[65754] = "xakwyrmezaj"; b[65754] =
a[65721] = "wabangrdan"; b[65721] =	"Ox1kwirmez1jaA0";
"Ow1b1ngard1nN1";	a[65755] = "xarpvstany"; b[65755] = "Ox1rpost1niX1";
a[65722] = "wagrdangyy"; b[65722] =	a[65756] = "xavrmyanhy"; b[65756] =
"Ow1gerd1negiiX1";	"Ox1varmiy1neiXu";
a[65723] = "wahdangany"; b[65723] =	"xavrwnasyy"; b[65757] =
"Ow1hd1neg1niX1";	"Ox1varwen1siiX1";
a[65724] = "waxkdarany"; b[65724] =	a[65758] = "xavyargry"; b[65758] = "Ox1viy1rgiriN1";
"Ow1xakd1r1niX1";	a[65759] = "xavyarsazy"; b[65759] = "Ox1viy1rs1ziN1";
a[65725] = "wearnvysyy"; b[65725] =	a[65760] = "xa~rxvahy"; b[65760] = "Ox1terx1hiiX1";
"Owo1rnevisiiX1";	a[65761] = "xbrpraknyy"; b[65761] =
a[65726] = "wearprdazy"; b[65726] =	"Oxabarpar1kaniiX1";
"Owo1rpard1ziN1";	a[65762] = "xbrprdazyy"; b[65762] =
a[65727] = "werbrnykhy"; b[65727] =	"Oxabarpard1ziiX1";
"OwarebrenikeiX1";	a[65763] = "xdabyamrzy"; b[65763] =
a[65728] = "wevr1fryny"; b[65728] = "Owoor1fariniN1";	"Oxod1biy1morziN1";
a[65729] = "whadtnamhy"; b[65729] =	a[65764] = "xdanwnasyy"; b[65764] =
"Owah1datn1meiX1";	"Oxod1nawn1siiX1";
a[65730] = "whbazxanyy"; b[65730] =	a[65765] = "xdanxvasth"; b[65765] = "Oxod1nax1stePa";
"Owahb1zx1niiXo";	a[65766] = "xdapsndanh"; b[65766] =
a[65731] = "whbazzadhy"; b[65731] =	"Oxod1pasand1neA0";
"Owahb1zz1deiXo";	a[65767] = "xdavndgara"; b[65767] =
a[65732] = "whrvrvstay"; b[65732] = "Owahrorst1iX1";	"Oxod1vandg1r1Pa";
a[65733] = "whryvrmahy"; b[65733] =	a[65768] = "xdavndgary"; b[65768] =
"Owahrivarm1hiX1";	"Oxod1vandeg1riX1";
a[65734] = "wkvfasazyy"; b[65734] =	a[65769] = "xmyrdndany"; b[65769] =
"Owokof1s1ziiX1";	"Oxamirdand1niX1";
a[65735] = "wmedanyany"; b[65735] =	a[65770] = "xnzrpnzryy"; b[65770] =
"Owamd1niy1niX1";	"OxenzerpenzeriiX1";
a[65736] = "wmwyrbazyy"; b[65736] =	a[65771] = "xrabkaranh"; b[65771] =
"Owamwirb1ziiX1";	"Oxar1bk1r1neA0";
a[65737] = "wmwymmahyy"; b[65737] =	a[65772] = "xrdgrayanh"; b[65772] =
"Owamwirm1hiiX1";	"Oxeradger1y1neAp";
a[65738] = "wnavrsazyy"; b[65738] =	a[65773] = "xrdsazvarh"; b[65773] = "Oxords1zv1reN1";
"Owen1vars1ziiX1";	a[65774] = "xrdvxakwyr"; b[65774] =
a[65739] = "wndrvpndry"; b[65739] =	"Oxordox1kwirA0";
"OwenderopenderiX1";	a[65775] = "xrdzystemd"; b[65775] =
a[65740] = "wryetmdary"; b[65740] =	"OxordzistmandN1";
"Owariatmad1riNo";	a[65776] = "xrsndkndndh"; b[65776] =
a[65741] = "wtrgavplng"; b[65741] =	"OxorsandkonandeA0";
"Owotorg1vpalangN1";	a[65777] = "xr~vmdaran"; b[65777] =
a[65742] = "wtrmr^kany"; b[65742] =	"Oxortomd1r1nN1";
"Owotormorqak1niX1";	a[65778] = "xsrvlbadyy"; b[65778] =
a[65743] = "wtrqrbanyy"; b[65743] =	"Oxosrov1b1diiXo";
"Owotorqorb1niiX1";	a[65779] = "xsrvprvyzy"; b[65779] =
	"OxosroparviziXa";

a[65780] = "xvahrzadhy"; b[65780] = "Ox1harz1deiX1";
a[65781] = "xvarzwmwahy"; b[65781] = "Ox1razmw1hiNo";
a[65782] = "xvastgaryy"; b[65782] = "Ox1steg1riiX1";
a[65783] = "xvd1gahanh"; b[65783] = "Oxod1g1h1neAp";
a[65784] = "xvd1zarndh"; b[65784] = "Oxod1z1randeA0";
a[65785] = "xvd1zmayyy"; b[65785] = "Oxod1zm1yiiX1";
a[65786] = "xvdalqayyy"; b[65786] = "Oxodelq1yiiX1";
a[65787] = "xvdatkaayy"; b[65787] = "Oxodetek1iiX1";
a[65788] = "xvdatkayyy"; b[65788] = "Oxodettek1yiiX1";
a[65789] = "xvdawt^aly"; b[65789] = "Oxodewteq1liiN1";
a[65790] = "xvdbalndgy"; b[65790] = "Oxodb1landegiN1";
a[65791] = "xvdbaxtgyy"; b[65791] = "Oxodb1xtegiX1";
a[65792] = "xvdbsndgyy"; b[65792] = "OxodbasandegiiX1";
a[65793] = "xvdbygangy"; b[65793] = "Oxodbig1negiN1";
a[65794] = "xvddrmany"; b[65794] = "Oxoddarm1niiX1";
a[65795] = "xvddryaby"; b[65795] = "Oxoddary1biiX1";
a[65796] = "xvdg@wtgyy"; b[65796] = "OxodgozawtegiX1";
a[65797] = "xvdgrdany"; b[65797] = "Oxodgard1niiX1";
a[65798] = "xvdmxtary"; b[65798] = "Oxodmox1riiX1";
a[65799] = "xvdprdaxy"; b[65799] = "Oxodpard1xtiX1";
a[65800] = "xvdpsndanh"; b[65800] = "Oxodpasand1neAp";
a[65801] = "xvdrdvzyly"; b[65801] = "OxodrodizeliX1";
a[65802] = "xvdrvsazy"; b[65802] = "Oxodros1ziiX1";
a[65803] = "xvdstrvny"; b[65803] = "OxodsetarvaniiX1";
a[65804] = "xvdtbnbyhy"; b[65804] = "OxodtanbihiiX1";
a[65805] = "xvdtxrybyy"; b[65805] = "OxodtaxribiiX1";
a[65806] = "xvdwkvfayy"; b[65806] = "Oxodwokof1yiN1";
a[65807] = "xvdwyftgyy"; b[65807] = "OxodwiftegiX1";
a[65808] = "xvdwrynyy"; b[65808] = "OxodwiriniiX1";
a[65809] = "xvdxvahanh"; b[65809] = "Oxodx1h1neAp";
a[65810] = "xvrwydyhay"; b[65810] = "Oxorwidih1iX1";
a[65811] = "xvwamdgvy"; b[65811] = "Oxow1madguiN1";
a[65812] = "xvwamdgvy"; b[65812] = "Oxow1madguyiN1";
a[65813] = "xvwnvysany"; b[65813] = "Oxownevis1niX1";
a[65814] = "xvywavndyy"; b[65814] = "Oxiw1vandiiX1";
a[65815] = "xvywtndary"; b[65815] = "Oxiwtand1riN1";
a[65816] = "xwayarwahy"; b[65816] = "Oxaw1y1rw1hiXa";
a[65817] = "xwnvdknndh"; b[65817] = "OxownodkonandeA0";
a[65818] = "xwnvdsazy"; b[65818] = "Oxownods1ziiX1";
a[65819] = "xyalprvryy"; b[65819] = "Oxyi1lparvariiX1";
a[65820] = "xyralnsaay"; b[65820] = "Oxeyronnes1liXa";
a[65821] = "xyrandywy"; b[65821] = "OxeyrandiwiiX1";
a[65822] = "xyrhhkndh"; b[65822] = "OxirekonandeA0";
a[65823] = "xyrxvahanh"; b[65823] = "Oxeyrx1h1neAp";
a[65824] = "yardanqlyy"; b[65824] = "Oy1rd1nqoliiX1";
a[65825] = "yasrerfaty"; b[65825] = "Oy1seraraf1tiXa";
a[65826] = "yhvdstyzyy"; b[65826] = "OyahudsetiziiX1";
a[65827] = "yksvknndhy"; b[65827] = "OyekasukonandeiX1";
a[65828] = "yvnyvrsaly"; b[65828] = "Oyunivers1liX1";
a[65829] = "yvnyvrsyty"; b[65829] = "OyuniversitiN1";
a[65830] = "yvnyzasyvn"; b[65830] = "Oyoniz1siyonN1";
a[65831] = "zalvsanany"; b[65831] = "Oz1los1niX1";
a[65832] = "zarzvndgyy"; b[65832] = "Oz1rozendegiiX1";
a[65833] = "znbazwkafy"; b[65833] = "Ozanb1zwek1fiN1";
a[65834] = "znbvrdary"; b[65834] = "Ozanburd1riiX1";
a[65835] = "znbvrstany"; b[65835] = "Ozanburest1niXu";
a[65836] = "znbvrxyvry"; b[65836] = "Ozanborx1riX1";
a[65837] = "zndanbany"; b[65837] = "Ozend1nb1niiX1";
a[65838] = "zndgynamhy"; b[65838] = "Ozendegin1meiX1";
a[65839] = "znjbylyany"; b[65839] = "Ozanjabiliy1niX1";
a[65840] = "znjyrbafyy"; b[65840] = "Ozanjirb1fiiX1";
a[65841] = "zvr1zmaayy"; b[65841] = "Ozur1zm1iiX1";
a[65842] = "zvr1zmayyy"; b[65842] = "Ozur1zm1yiiX1";
a[65843] = "zyadxvahy"; b[65843] = "Oziy1dx1hiiX1";
a[65844] = "zyartnamhy"; b[65844] = "Oziy1ratn1meiX1";
a[65845] = "zybapsndyy"; b[65845] = "Ozib1pasandiiX1";
a[65846] = "zybawnasy"; b[65846] = "Ozib1wen1siiX1";
a[65847] = "zybawnaxy"; b[65847] = "Ozib1wen1xtiN1";
a[65848] = "zyrdbstany"; b[65848] = "Ozirdabest1niA0";
a[65849] = "zyrdryaayy"; b[65849] = "Ozirdary1iiX1";
a[65850] = "zyrdryayyy"; b[65850] = "Ozirdary1yiiX1";
a[65851] = "zymjmvvhy"; b[65851] = "OzirmajmueiX1";
a[65852] = "zyrpyrahny"; b[65852] = "Ozirpir1haniX1";
a[65853] = "zyrwyrvany"; b[65853] = "Ozirwirv1niN1";
a[65854] = "zyrxanvadh"; b[65854] = "Ozix1nev1deN1";

```

}

private void richTextBox1_TextChanged(object sender,
EventArgs e)
{
    nword=richTextBox1.Text;
    //cancel(" "+nword);
}
public void cancel(string xword)
{
    string xx1, xx2, xx3, xx4, xx5, yy1="", yy2="", yy3="",
yy4="", yy5="", numout = "O-";
    string[] outtext = new string[70000]; char[] ychar = new
char[10000];
    xword = xword.Replace(" ", " ");
    xword = xword.Replace(" ", " ");
    xword = xword.Replace(" ", " ");
    xword = xword.Replace(" ", " ");
    xword = xword.Replace(" \\r\\n", " ");
    xword=xword.Replace(", ", " ");
    xword=xword.Replace(";", " ");
    xword=xword.Replace(":", " ");
    xword=xword.Replace(".", " ");
    xword=xword.Replace("/", " ");
    xword=xword.Replace("\\", " ");
    xword=xword.Replace("(", " ");
    xword=xword.Replace(")", " ");
    xword=xword.Replace(":", " ");
    xword=xword.Replace("[", " ");
    xword=xword.Replace("]", " ");
    xword=xword.Replace("{", " ");
    xword=xword.Replace("}", " ");
    xword=xword.Replace("!", " ");
    xword=xword.Replace("?", " ");
    xword=xword.Replace(">", " ");
    xword=xword.Replace("<", " ");
    xword=xword.Replace("@", " ");
    xword=xword.Replace("#", " ");
    xword=xword.Replace("$", " ");
    xword=xword.Replace("%", " ");
    xword=xword.Replace("^", " ");
    xword=xword.Replace("&", " ");
    xword=xword.Replace("*", " ");
    xword=xword.Replace("-", " ");
    xword=xword.Replace("=", " ");
    xword=xword.Replace("_", " ");
    xword=xword.Replace("+", " ");
    string[] xxword=xword.Split(' ');
    ixword = xxword.Count();
    string outsentece="";
    string outout = "";
    string[] xoutext=new string[70000];
    int[] issame=new int[70000];
    string[] grammer = new string[70000]; string[] word = new
string[70000];
    word[0] = "----";
    grammer [0]="--";
    int eh=0;

    for (int i = 1; i < ixword; i++)
    {

```

```

if (xxword[i].Length > 0)
{
    char[] xchar = xxword[i].ToArray();
    int[] valuexchar = new int[xxword[i].Length];
    outtext[i] = "";
    grammer[i] = "--";

    eh = 0;
    for (int j = 0; j < xxword[i].Length; j++)
    {
        valuexchar[j] = xchar[j].GetHashCode();
        if (xchar[j] == '1') ychar[j] = '*';
        else
        {
            if (xchar[j] == '2') ychar[j] = '2';
            else
            {
                if (xchar[j] == '3') ychar[j] = '3';
                else
                {
                    if (xchar[j] == '4') ychar[j] = '4';
                    else
                    {
                        if (xchar[j] == '5') ychar[j] = '5';
                        else
                        {
                            if (xchar[j] == '6') ychar[j] = '6';
                            else
                            {
                                if (xchar[j] == '7') ychar[j] = '7';
                                else
                                {
                                    if (xchar[j] == '8') ychar[j] = '8';
                                    else
                                    {
                                        if (xchar[j] == '9') ychar[j] =
'9';
                                        else
                                        {
                                            if (xchar[j] == '0') ychar[j]
= '0';
                                            else
                                            {
                                                if (xchar[j] == 'ض')
ychar[j] = '#';
                                                else
                                                {
                                                    if (xchar[j] == 'ص')
ychar[j] = '$';
                                                    else
                                                    {
                                                        if (xchar[j] == 'ث')
ychar[j] = '%';
                                                        else
                                                        {
                                                            if (xchar[j] ==
'ق') ychar[j] = 'q';
                                                            else
                                                            {

```



```

'ف') ychar[j] = 'f';

== 'غ') ychar[j] = '^';

(xchar[j] == 'ع') ychar[j] = 'e';

(xchar[j] == 'ع') ychar[j] = 'e';

(xchar[j] == 'ه') ychar[j] = 'h';

(xchar[j] == 'خ') ychar[j] = 'x';

else

if (xchar[j] == 'ح') ychar[j] = '&';

else

{

if (xchar[j] == 'ج') ychar[j] = 'j';

else

{

if (xchar[j] == 'چ') ychar[j] = 'c';

else

{

if (xchar[j] == 'پ') ychar[j] = 'p';

else

{

if (xchar[j] == 'ش') ychar[j] = 'w';

else

{

if (xchar[j] == 'س') ychar[j] = 's';

else

{

if (xchar[j] == 'ی') ychar[j] = 'y';

else

{

if (xchar[j] == 'ی') ychar[j] = 'y';

else

{

if (xchar[j] == 'ی') ychar[j] = 'y';

else

{

if (xchar[j] == 'ب') ychar[j] = 'b';

else

{

if (xchar[j] == 'ل') ychar[j] = 'l';

else

{

if (xchar[j] == 'ا') ychar[j] = 'a';

else

{

if (xchar[j] == 'ت') ychar[j] = 't';

else

{

if (xchar[j] == 'ن') ychar[j] = 'n';

else

{

if (xchar[j] == 'م') ychar[j] = 'm';

else

{

if (xchar[j] == 'ك') ychar[j] = 'k';

else

{

```

```
if (xchar[j] == 'ك') ychar[j] = 'k';
else
{
if (xchar[j] == 'گ') ychar[j] = 'g';
else
{
if (xchar[j] == 'ط') ychar[j] = 't';
else
{
if (xchar[j] == 'ب') ychar[j] = 'b';
else
{
if (xchar[j] == 'ج') ychar[j] = 'z';
else
{
if (xchar[j] == 'د') ychar[j] = 'r';
else
{
if (xchar[j] == 'ذ') ychar[j] = '@';
else
{
if (xchar[j] == 'س') ychar[j] = 'd';
else
{
if (xchar[j] == 'ل') ychar[j] = 'l';
else
{
if (xchar[j] == 'ش') ychar[j] = '-';
else
{
```



```

    }
    if (xx1.Length > 0)
    {
        if (xx1.Substring(0, 1) == "0")
        {
            xx1 = xx1.Substring(1, xx1.Length - 1);
            yy1 = "sefr" + yy1;
        }
    }
}

yy1 = yy1+Search3Number(xx1);
xx2 = outtext[i].Substring(0, outtext[i].Length -
3);

if (xx2.Length > 3)
{
    xx3 = xx2.Substring(xx2.Length - 3, 3);
    - 1);

    if (telno == 1)
    {
        if (xx3.Length > 0)
        {
            if (xx3.Substring(0, 1) == "0")
            {
                xx3 = xx3.Substring(1, xx3.Length -
1);
                yy3 = "sefr";
            }
        }
        if (xx3.Length > 0)
        {
            if (xx3.Substring(0, 1) == "0")
            {
                xx3 = xx3.Substring(1, xx3.Length -
1);
                yy3 = "sefr" + yy3;
            }
        }
        if (xx3.Length > 0)
        {
            if (xx3.Substring(0, 1) == "0")
            {
                xx3 = xx3.Substring(1, xx3.Length -
1);
                yy3 = "sefr" + yy3;
            }
        }
    }

    yy3 = yy3+Search3Number(xx3);
    xx4 = xx2.Substring(0, xx2.Length - 3);
    if (xx4.Length > 3)
    {
        xx5 = xx4.Substring(xx4.Length - 3, 3);
        - 1);
    }

    if (telno == 1)
    {
        if (xx5.Length > 0)
        {
            if (xx5.Substring(0, 1) == "0")
            {
                xx5 = xx5.Substring(1, xx5.Length
- 1);
                yy5 = "sefr";
            }
        }
        if (xx5.Length > 0)
        {
            if (xx5.Substring(0, 1) == "0")
            {
                xx5 = xx5.Substring(1, xx5.Length
- 1);
                yy5 = "sefr" + yy5;
            }
        }
        if (xx5.Length > 0)
        {
            if (xx5.Substring(0, 1) == "0")
            {
                xx5 = xx5.Substring(1, xx5.Length
- 1);
                yy5 = "sefr" + yy5;
            }
        }
    }

    yy5 = yy5+Search3Number(xx5);
    //xx6 = outtext[i].Substring(3,
outtext[i].Length - 3);
    //yy6=Search3Number(xx6);
    if (telno==1)
        numout = yy5 + yy3 + yy1;
    else
        numout = yy5 + "milyono" + yy3 +
"hez1ro" + yy1;
    else
    {
        if (telno == 1)
        {
            if (xx4.Length > 0)
            {
                if (xx4.Substring(0, 1) == "0")
                {
                    xx4 = xx4.Substring(1, xx4.Length
- 1);
                    yy4 = "sefr";
                }
            }
            if (xx4.Length > 0)

```

```

        {
            if (xx4.Substring(0, 1) == "0")
            {
                xx4 = xx4.Substring(1, xx4.Length
- 1);
                yy4 = "sefr" + yy4;
            }
        }
        if (xx4.Length > 0)
        {
            if (xx4.Substring(0, 1) == "0")
            {
                xx4 = xx4.Substring(1, xx4.Length
- 1);
                yy4 = "sefr" + yy4;
            }
        }
        yy4 = yy4+Search3Number(xx4);
        if (telno == 1)
            numout = yy4 + yy3 + yy1;
        else
            numout = yy4 + "milyono" + yy3 +
"hez1ro" + yy1;
    }
    else
    {
        if (telno == 1)
        {
            if (xx2.Length > 0)
            {
                if (xx2.Substring(0, 1) == "0")
                {
                    xx2 = xx2.Substring(1, xx2.Length -
1);
                    yy2 = "sefr";
                }
            }
            if (xx2.Length > 0)
            {
                if (xx2.Substring(0, 1) == "0")
                {
                    xx2 = xx2.Substring(1, xx2.Length -
1);
                    yy2 = "sefr" + yy2;
                }
            }
            if (xx2.Length > 0)
            {
                if (xx2.Substring(0, 1) == "0")
                {
                    xx2 = xx2.Substring(1, xx2.Length -
1);
                    yy2 = "sefr" + yy2;
                }
            }
        }
        yy2 = Search3Number(xx2);
        if (telno == 1)
            numout = yy2 + yy1;
        else
            numout = yy2 + "hez1ro" + yy1;
    }
    else
    {
        if (telno == 1)
        {
            if (outtext[i].Length > 0)
            {
                if (outtext[i].Substring(0, 1) == "0")
                {
                    outtext[i] = outtext[i].Substring(1,
outtext[i].Length - 1);
                    numout = "sefr";
                }
            }
            if (outtext[i].Length > 0)
            {
                if (outtext[i].Substring(0, 1) == "0")
                {
                    outtext[i] = outtext[i].Substring(1,
outtext[i].Length - 1);
                    numout = "sefr" + numout;
                }
            }
            if (outtext[i].Length > 0)
            {
                if (outtext[i].Substring(0, 1) == "0")
                {
                    outtext[i] = outtext[i].Substring(1,
outtext[i].Length - 1);
                    numout = "sefr" + numout;
                }
            }
        }
        numout = numout +Search3Number(outtext[i]);
        word[i] = numout;
    }
    else
    {

```



```

//else
//mx4=StringReverse(StringSegment(ggoo, " ",mi*(-1)))
//;
//mi=i+2
//if StringLeft(StringReverse(StringSegment(GGoo, " ",mi*(-1))),1)=="N"
StringLeft(StringReverse(StringSegment(mmoo, " ",mi*(-1))),1)=="V"
//mx5=StringReverse(StringSegment(mmoo, " ",mi*(-1)))
//else
//mx5=StringReverse(StringSegment(ggoo, " ",mi*(-1)))
//;
//mi=i+3
//if StringLeft(StringReverse(StringSegment(GGoo, " ",mi*(-1))),1)=="N"
StringLeft(StringReverse(StringSegment(mmoo, " ",mi*(-1))),1)=="V"
//mx6=StringReverse(StringSegment(mmoo, " ",mi*(-1)))
//else
//mx6=StringReverse(StringSegment(ggoo, " ",mi*(-1)))
//;
//mi=i+4
//if StringLeft(StringReverse(StringSegment(GGoo, " ",mi*(-1))),1)=="N"
StringLeft(StringReverse(StringSegment(mmoo, " ",mi*(-1))),1)=="V"
//mx7=StringReverse(StringSegment(mmoo, " ",mi*(-1)))
//else
//mx7=StringReverse(StringSegment(ggoo, " ",mi*(-1)))
//;
//if mx1=="N1" && StringRight(nn0,1)=="h"
//mx1="M1"
//;
//if mx3=="N1" && StringRight(n2,1)=="h"
//mx3="M1"
//;

//if kasreh(ee,i-1,mx1,mx2,mx3,mx4,mx5,mx6,mx7) ;;&&
AKasrehx(n1) && AKasreh(n2)
//;;;;;;;; show(n0+" "+mx1+" "+mx2+" "+mx3+" "+mx4+"
"+mx5+" "+mx6+" "+mx7)
//;;;;;;;; Kasreh ;;;show(mx2+" "+mx3+" "+mx4+" "+mx5+"
"+mx6+" "+n0+" 1 Before "+mx1)

//if StringRight(n0,1)=="e"
//n0=n0+"ye"
//else
//if StringRight(n0,1)=="i" || StringRight(n0,1)=="1"
//else
//n0=n0+"e"
//;
//;
//;
//;
//GloTextToWav=GloTextToWav+" "+n0
//EndWhile
//Endfunction
}
public int kasreh(int end, int nomre, string e0, string e1,
string e2, string e3, string e4, string e5, string e6)
{
int x1 = 0, x2 = 0;
if (e1 == null) e1 = "--";
if (e2 == null) e2 = "--";
if (e3 == null) e3 = "--";
if (e4 == null) e4 = "--";
if (e5 == null) e5 = "--";
if (e6 == null) e6 = "--";

// added 1393
if ((e1 == "N9" && e2 == "Ne") || (e1 == "N1" && e2
== "Ne") || (e1 == "A0" && e2 == "Ne") || (e1 == "N1" && e2
== "Ne") || (e1 == "A0" && e2 == "Ne")) return 1;

if (e1 == "A0" && e2 == "A1") return 0;
if (e1 == "N1" && e2 == "N1" && (e3 == "P3" || e3
== "XX" || e3 == "V3" || e3 == "V1" || e3 == "A2" || e3 == "VV"
|| e3 == "C0")) return 0;
if (e0 == "Eh" && e1 == "N1" && e2 == "N1") return
0;
if (e0 == "N1" && e1 == "N1" && e2 == "N1") return
0;
if (e0 == "P3" && e1 == "N1" && e2 == "A1") return
0;
if (e0 == "A1" && e1 == "N1" && e2 == "A0") return
0;
if (e1 == "N1" && e2 == "A0" && e3 == "V1") return
0;
if (e1 == "N1" && e2 == "A2") return 0;
if (e1 == "A0" && e2 == "N1") return 0;
if (e0.Substring(0, 1) == "V") return 0;
if (e0 == "Po" || e2 == "Po" || e2 == "Pr" || e2 == "Pt")
return 0;
if (e1 == "A0" && e2.Substring(0, 1) == "V") return
0;
if (end == nomre) return 0;
if (e1 == "Nu" && e2.Substring(0, 1) == "N") return
0;

if (e1.Length > 0)
{
if (e1.Length > 1 && (e1.Substring(1, 1) == "0" ||
e1.Substring(1, 1) == "1" || e1.Substring(1, 1) == "2" ||
e1.Substring(1, 1) == "3" || e1.Substring(1, 1) == "4" ||
e1.Substring(1, 1) == "5" || e1.Substring(1, 1) == "6" ||
e1.Substring(1, 1) == "7" || e1.Substring(1, 1) == "8" ||
e1.Substring(1, 1) == "9"))
x1 = Convert.ToInt32(e1.Substring(1, 1));
else
x1 = 0;
}
else x1 = 0;
if (e2.Length > 0)
{
if (e2.Length > 1 && (e2.Substring(1, 1) == "0" ||
e2.Substring(1, 1) == "1" || e2.Substring(1, 1) == "2" ||
e2.Substring(1, 1) == "3" || e2.Substring(1, 1) == "4" ||
e2.Substring(1, 1) == "5" || e2.Substring(1, 1) == "6" ||
e2.Substring(1, 1) == "7" || e2.Substring(1, 1) == "8" ||
e2.Substring(1, 1) == "9"))
x2 = Convert.ToInt32(e2.Substring(1, 1));
else
}
}

```

```

        x2 = 0;
    }
    else x2 = 0;
    if (e1.Substring(0, 1) == "N" && e2.Substring(0, 1) ==
"N" && x1 < x2) return 1;
    if (e1.Substring(0, 1) == "N" && e2.Substring(0, 1) ==
"N" && x1 > x2) return 0;

    if (e0.Substring(0, 1) == "N" && e1 == "A0" &&
e2.Substring(0, 1) == "N") return 1;
    if ((e0 == "Pr" || e0 == " ") && e1 == "A0" && e2 ==
"N1") return 0;

    if (e1 == "N1" && e2 == "N1" && (e3 == "P3" || e3
== "XX" || e3 == "V3" || e3 == "V1" || e3 == "A2" || e3 == "VV"
|| e3 == "C0")) return 0;
    if (e1 == "N1" && e2 == "N1" && (e3 == "V2" ||
e3 == "Vh")) return 0;
    if (e1 == "V1" || e1 == "V2" || e1 == "V3" || e3 ==
"V3") return 0;

    if (e1 == "N1" && e2 == "A0" && e3 == "N1"
&& e4 == "V3") return 0;

    if ((e1 == "N1" && e2 == "N1" && e3 == "V2"
&& e4 == "V3" && (e0 != "Pr")) || (e1 == "N1" && e2 == "A0"
&& e3 == "C0" && (e0 != "V1"))) || (e1 == "N1" && (e3 ==
"V2" || e3 == "V1" || e3 == "Pr") && e2 == "N1" && (!(e0 ==
"Ny")) && (!(e0 == "A1")) && (!(e0 == "SS"))) || (e1 == "N1"
&& e2 == "A0" && e3 == "Si") || (e1 == "N1" && e2 == "N1"
&& e0 == "Si") || (e1 == "A0" && e2 == "N1" &&
(e3.Substring(0, 1) == "V" || e3 == "N1" || e3 == "N5" || e3 ==
"C0") || e3 == "Pr")) || (e1 == "V1" && e2 == "N1" && (!(e0 ==
"A0")) && (e3 == "N1" || e3 == "V1")) || (e1 == "N1" && e2 ==
"N1" && e3 == "A0" && e4 == "A0" && e5 == "V3" && (e0
!= "Pr")) || (e1 == "N1" && e2 == "N1" && e3 == "N5" && e0
== "Nm") || (e1 == "N1" && e2 == "N1" && e3 == "N3" &&
e4 == "Po" && e0 == "Pr")) || (e1 == "N1" && e2 == "N1" &&
e3 == "A0" && (e4 != "V1") && e0 == "Px") || (e1 == "N1"
&& e2 == "N3" && e3 == "A0" && e0 == "A1") || (e1 == "N1"
&& e2 == "A0" && e3 == "V1" && e0 == "Nm") || (e1 ==
"N1" && e2 == "A0" && e3 == "V2" && e0 == "Pr") || (e1 ==
"N1" && e2 == "A0" && e3 == "V1" && e0 == "Px") || (e1 ==
"N1" && e2 == "V1" && (e3 == "Pr" || e3 == "V1" || e3 ==
"SS")) || (e1 == "N1" && e2 == "V1" && (e0 != "Pr") && e3
== "C0") || (e1 == "N1" && e2 == "V1" && e3 == "A0" && e4
== "N5") || (e1 == "N1" && e2 == "V1" && (e3 != "C0") &&
(e0 == "Nx" || e0 == "SS" || e0 == "N3" || e0 == "Pr")) || (e1 ==
"N1" && e2 == "V1" && (e3 != "N1") && e0 == "Pr") || (e1 ==
"A0" && e2 == "A0" && e3 == "SS" && e0 == "Nx") || (e1 ==
"A0" && e2 == "N1" && e3 == "Px" && (e4 == "Nx" || e4 ==
"N1")) && e0 == "N1") || (e1 == "V2" && e2 == "Pr") || (e1 ==
"Nm" && e2 == "N1") || (e1 == "N1" && e2 == "Nm" && e0
== "Pr") || (e1 == "N1" && e2 == "N1" && e3 == "N5" && e0
== "C0") || (e1 == "N1" && e2 == "V1" && e3 == "A0" && e4
== "V3") || (e1 == "Ad" && e2 == "Ad" && e0 == "Si") || (e1
== "V2" && e2 == "A0") || (e1 == "A0" && e2 == "V1" && e0
== "V2") || (e1 == "N1" && e2 == "V1" && e3 == "V3") || (e1
== "N1" && e2 == "N1" && e3 == "KK") || (e1 == "N1" && e2

```

```

== "A0" && e3 == "V3") || (e1 == "A0" && e2 == "A0" &&
(e3 == "V3" || e3 == "Va")) || (e1 == "N1" && e2 == "N1" &&
e3 == "V3") || (e1 == "N1" && e2 == "N1" && e3 == "Po") ||
(e1 == "N1" && e2 == "V1") || (e1 == "N1" && e2 == "A0"
&& e3 == "A0" && e4 == "V3")) return 0;

```

```

    if (e1 == "N9" && e2 == "N5") || (e1 == "N1"
&& e2 == "N5") || (e1 == "A0" && e2 == "N5") || (e1 == "N1"
&& e2 == "N3") || (e1 == "A0" && e2 == "N3") || (e1 == "N1"
&& e2 == "A1") || (e1 == "N1" && e2 == "Ad") || (e1 == "A0"
&& e2 == "A1") || (e1 == "V2" && e2 == "N5") || (e1 == "V2"
&& e2 == "N3") || (e1 == "A0" && e2 == "N7") || (e1 == "A0"
&& e2 == "N5") || (e1 == "A0" && e2 == "N3") || (e1 == "A0"
&& e2 == "N1" && (e3.Substring(0, 1) != "V" || e3 != "C0")) ||
(e1 == "A0" && e2 == "N2") || (e1 == "N2" && e2 == "A0") ||
(e1 == "A1" && e2 == "N7") || (e1 == "A1" && e2 == "N8") ||
(e1 == "V2" && e2 == "A0") || (e1 == "N1" && e2 == "N7") ||
(e1 == "A0" && e2 == "A0") || (e1 == "V1" && e2 == "N5") ||
(e1 == "V2" && e2 == "Pr") || (e1 == "N1" && e2 == "Nm") ||
(e1 == "N1" && e2 == "N5") || (e1 == "N1" && e2 == "A2") ||
(e1 == "N3" && e2 == "N4") || (e1 == "Z1" && e2 == "N1") ||
(e1 == "N1" && e2 == "A0") || (e1 == "N1" && e2 == "N5") ||
(e1 == "N1" && e2 == "N4") || (e1 == "N1" && e2 == "V1"
&& e3 != "Si") || (e1 == "V1" && e2 == "N1") || (e1 == "A0"
&& e2 == "N5") || (e1 == "A0" && e2 == "N3") || (e1 == "N1"
&& e2 == "N1" && e3 == "A0") || (e1 == "N1" && e2 == "N1"
&& e3 == "V1") || (e1 == "N1" && e2 == "A0" && (e3 ==
"V1" || e3 == "A0" || e3 == "V3" || e3 == "V2")) || (e1 == "N1"
&& e2 == "A0" && e3 == "A0" && e4 == "V3") || (e1 == "N1"
&& e2 == "N1" && e3 == "A0") || (e1 == "A0" && e2 ==
"N5") || (e1 == "A0" && e2 == "N3") || (e1 == "N1" && e2 ==
"N3") || (e1 == "N1" && e2 == "N5") || (e1 == "N1" && e2 ==
"N7") || (e1 == "N9" && e2 == "N1") || (e1 == "N9" && e2 ==
"N3") || (e1 == "N9" && e2 == "N2") || (e1 == "N1" && e2 ==
"N9") || (e1 == "N1" && e2 == "Nx") || (e1 == "Ny" && e2 ==
"N1") || (e1 == "A0" && e2 == "A0") || (e1 == "Ad" && e2 ==
"Ad") || (e1 == "A0" && e2 == "Ab") || (e1 == "V2" && e2 ==
"N5") || (e1 == "A0" && e2 == "N7") || (e1 == "A0" && e2 ==
"N1" && e3.Substring(0, 1) == "V") || (e1 == "N1" && e2 ==
"A0") || (e1 == "V1" && e2 == "N1" && e0 == "A0") || (e1 ==
"N1" && e2 == "N1" && e0 == "Pr" && (e3 == "V2" || e3 ==
"V1")) || (e1 == "Nz" && e2 == "Np") || (e1 == "Nz" && e2 ==
"N5") || (e1 == "Ny" && e2 == "Ad") || (e1 == "Ny" && e2 ==
"A0") || (e1 == "N1" && e2 == "Np") || (e1 == "Np" && e2 ==
"N1") || (e1 == "N1" && e2 == "V1" && e0 == "Pr" && (e3 ==
"C0" || e3 == "N1")) || (e1 == "N1" && e2 == "N1" && e0 ==
"Pr" && (e3 == "V2" || e3 == "V1")) || (e1 == "V1" && e2 ==
"N1" && e0 == "A0") || (e1 == "N1" && e2 == "A0" && e0 ==
"V1" && e3 == "N1" && e4 == "Px") || (e1 == "N1" && e2 ==
"N1" && e0 == "Pr" && e2 == "A0") || (e1 == "N1" && e2 ==
"N1" && e0 == "Ny" && e2 == "Pr") || (e1 == "N1" && e2 ==
"A0" && e0 == "V1" && e2 == "N1") || (e1 == "Np" && e2 ==
"N5") || (e1 == "N1" && e2 == "A0" && e3 == "N3") || (e1 ==
"N1" && e2 == "A0" && e3 == "N1") || (e1 == "N1" && e2 ==
"N1")) return 1;

```

```

    if ((e1 == "V2" && e2 == "V1" && e3 == "C0"
&& e0 == "N1") || (e1 == "N1" && e2 == "N1" && e3 == "Pr"
&& e0 == "Pr") || (e1 == "N1" && e2 == "N1" && e3 == "V2"
&& e0 == "Pr" && e4 == "V1") || (e1 == "N1" && e2 == "V1"
&& e3 == "N1" && e0 == "Pr" && e4 == "A0") || (e1 == "N1"

```



```

&& e2 == "A0" && e3 == "N1" && e4 == "N5" && e5 ==
"V3") || (e1 == "N1" && e2 == "No")) return 1;

        return 0;
    }

    private void radioButton1_CheckedChanged(object sender,
EventArgs e)
    {
        if (radioButton1.Checked)
            voice = "M";
    }

    public void mbrola(string ss1)
    {

        //string Runit;
        ss1 = ss1.Replace("-", "");
        ss1 = ss1.Replace("aa", "1");
        ss1 = ss1.Replace(" 1a", " 1");
        ss1 = ss1.Replace("u1", "1");
        ss1 = ss1.Replace("C", "K");
        ss1 = ss1.Replace("W", "V");
        ss1 = ss1.Replace("Q", "q");
        ss1 = ss1.Replace("GH", "Q");
        ss1 = ss1.Replace("KH", "X");
        ss1 = ss1.Replace("ZH", "7");
        ss1 = ss1.Replace("SH", "W");

        //if dari==1 {
        //ss1 = ss1.Replace("e", "a");
        //}

        //;;;Transfer_M_mini(GloTextToWav+cocotext,i,PhoFile)

        //ss1=stringlower(ss1)
        //ss1=StringTrimTrailingBlanks(StringTrimLeadingBlanks(ss1))
        ss1 = ss1.Replace("iy1n", "i1n");
        ss1 = ss1.Replace("7", "|");
        ss1 = ss1.Replace("hj", "h6j");
        ss1 = ss1.Replace("000", "eye");
        ss1 = ss1.Replace("1", "%");
        ss1 = ss1.Replace("6", "8");
        ss1 = ss1.Replace("8", "^");
        ss1 = ss1.Replace("9", "*");
        ss1 = ss1.Replace("ej", "e^$");
        ss1 = ss1.Replace("j", "$");
        ss1 = ss1.Replace("og", "o^g");
        ss1 = ss1.Replace("h j", "h^j");
        ss1 = ss1.Replace("$ c", "j c");
        ss1 = ss1.Replace("$ g", "j g");
        ss1 = ss1.Replace("$ j", "j j");
        ss1 = ss1.Replace("$ h", "j h");
        ss1 = ss1.Replace("$ q", "j q");
        ss1 = ss1.Replace("$ p", "j p");
        ss1 = ss1.Replace("$ t", "j t");
        ss1 = ss1.Replace("$ v", "j v");

        ss1 = ss1.Replace("$ w", "j w");
        ss1 = ss1.Replace("$ y", "j y");
        ss1 = ss1.Replace("g 0", "# 0");
        ss1 = ss1.Replace("g a", "# a");
        ss1 = ss1.Replace("g i", "# i");
        ss1 = ss1.Replace("g e", "# e");
        ss1 = ss1.Replace("g 6", "# 6");
        ss1 = ss1.Replace("g 9", "# 9");
        ss1 = ss1.Replace("n j", "n $");
        ss1 = ss1.Replace("$c", "jc");
        ss1 = ss1.Replace("$g", "jg");
        ss1 = ss1.Replace("$j", "jj");
        ss1 = ss1.Replace("$h", "jh");
        ss1 = ss1.Replace("$q", "jq");
        ss1 = ss1.Replace("$p", "jp");
        ss1 = ss1.Replace("$t", "jt");
        ss1 = ss1.Replace("$v", "jv");
        ss1 = ss1.Replace("$w", "jw");
        ss1 = ss1.Replace("$y", "jy");
        ss1 = ss1.Replace("g0", "#0");
        //;ss1=StringReplaceSubstrings(ss1,"g%", "#%");
        ss1 = ss1.Replace("ga", "#a");
        ss1 = ss1.Replace("gi", "#i");
        ss1 = ss1.Replace("ge", "#e");
        ss1 = ss1.Replace("g6", "#6");
        ss1 = ss1.Replace("g9", "#9");
        ss1 = ss1.Replace("c", "@");
        ss1 = ss1.Replace("k", "c");
        ss1 = ss1.Replace("c%", "k%");
        ss1 = ss1.Replace("co", "ko");
        ss1 = ss1.Replace("cu", "ku");
        ss1 = ss1.Replace("c4", "k4");
        ss1 = ss1.Replace("c5", "k5");
        ss1 = ss1.Replace("c %", "k %");
        ss1 = ss1.Replace("c o", "k o");
        ss1 = ss1.Replace("c u", "k u");
        ss1 = ss1.Replace("c 4", "k 4");
        ss1 = ss1.Replace("c 5", "k 5");
        //;ss1=StringReplaceSubstrings(ss1,"g", "#");
        ss1 = ss1.Replace("nj", "n$");
        ss1 = ss1.Replace("uu", "u");
        ss1 = ss1.Replace("gy", "gi");
        ss1 = ss1.Replace("gv", "g7v");
        //;ss1=StringReplaceSubStrings(ss1,"g d", "g7d");
        //;ss1=StringReplaceSubStrings(ss1,"g l", "g7l");
        ss1 = ss1.Replace("gd", "g7d");
        ss1 = ss1.Replace("gl", "g7l");
        ss1 = ss1.Replace("gr", "g7r");
        ss1 = ss1.Replace("gz", "g7z");
        ss1 = ss1.Replace("g v", "g7 v");
        ss1 = ss1.Replace("g r", "g7 r");
        ss1 = ss1.Replace("g z", "g7 z");
        ss1 = ss1.Replace("%", "%");
        ss1 = ss1.Replace(" u", "u");
        ss1 = ss1.Replace(" a", "a");
        ss1 = ss1.Replace(" i", "i");
        ss1 = ss1.Replace(" o", "o");
        ss1 = ss1.Replace(" e", "e");
        ss1 = ss1.Replace("%%", "%^%");
        ss1 = ss1.Replace("uu", "u^u");
        ss1 = ss1.Replace("aa", "a^a");
    }

```

```

ss1 = ss1.Replace("ii", "i^i");
ss1 = ss1.Replace("oo", "o^o");
ss1 = ss1.Replace("ee", "e^e");
ss1 = ss1.Replace("uo", "u^o");
ss1 = ss1.Replace("ua", "u^a");
ss1 = ss1.Replace("ue", "u^e");
ss1 = ss1.Replace("ui", "u^i");
ss1 = ss1.Replace("u%", "u^%");
ss1 = ss1.Replace("ao", "a^o");
ss1 = ss1.Replace("au", "a^u");
ss1 = ss1.Replace("ae", "a^e");
ss1 = ss1.Replace("ai", "a^i");
ss1 = ss1.Replace("a%", "a^%");
ss1 = ss1.Replace("%o", "%^o");
ss1 = ss1.Replace("%u", "%^u");
ss1 = ss1.Replace("%e", "%^e");
ss1 = ss1.Replace("%i", "%^i");
ss1 = ss1.Replace("%a", "%^a");
ss1 = ss1.Replace("o%", "o^%");
ss1 = ss1.Replace("oa", "o^a");
ss1 = ss1.Replace("oe", "o^e");
ss1 = ss1.Replace("ou", "o^u");
ss1 = ss1.Replace("oi", "o^i");
ss1 = ss1.Replace("i%", "i^%");
ss1 = ss1.Replace("ia", "i^a");
ss1 = ss1.Replace("io", "i^o");
ss1 = ss1.Replace("iu", "i^u");
ss1 = ss1.Replace("ie", "i^e");
ss1 = ss1.Replace("e%", "e^%");
ss1 = ss1.Replace("ea", "e^a");
ss1 = ss1.Replace("eo", "e^o");
ss1 = ss1.Replace("eu", "e^u");
ss1 = ss1.Replace("ei", "e^i");
ss1 = ss1.Replace("hj", "h^j");
ss1 = ss1.Replace("$c", "j^c");
ss1 = ss1.Replace("$g", "j^g");
ss1 = ss1.Replace("$j", "j^j");
ss1 = ss1.Replace("$h", "j^h");
ss1 = ss1.Replace("$q", "j^q");
ss1 = ss1.Replace("$p", "j^p");
ss1 = ss1.Replace("$t", "j^t");
ss1 = ss1.Replace("$v", "j^v");
ss1 = ss1.Replace("$w", "j^w");
ss1 = ss1.Replace("$y", "j^y");
ss1 = ss1.Replace("g0", "#0");
ss1 = ss1.Replace("ga", "#a");
ss1 = ss1.Replace("gi", "#i");
ss1 = ss1.Replace("ge", "#e");
ss1 = ss1.Replace("g^", "#^");
ss1 = ss1.Replace("g9", "#9");
ss1 = ss1.Replace("c%", "k%");
ss1 = ss1.Replace("co", "ko");
ss1 = ss1.Replace("cu", "ku");
ss1 = ss1.Replace("^j", "^$");
ss1 = ss1.Replace("%%", "%");
ss1 = ss1.Replace("h j", "h^j");
ss1 = ss1.Replace("$ c", "j^c");
ss1 = ss1.Replace("$ g", "j^g");
ss1 = ss1.Replace("$ j", "j^j");
ss1 = ss1.Replace("$ h", "j^h");
ss1 = ss1.Replace("$ q", "j^q");

```

```

ss1 = ss1.Replace("$ p", "j^p");
ss1 = ss1.Replace("$ t", "j^t");
ss1 = ss1.Replace("$ v", "j^v");
ss1 = ss1.Replace("$ w", "j^w");
ss1 = ss1.Replace("$ y", "j^y");
ss1 = ss1.Replace("g 0", "# 0");
ss1 = ss1.Replace("g a", "# a");
ss1 = ss1.Replace("g i", "# i");
ss1 = ss1.Replace("g e", "# e");
ss1 = ss1.Replace("g ^", "# ^");
ss1 = ss1.Replace("g 9", "# 9");
ss1 = ss1.Replace("c %", "k %");
ss1 = ss1.Replace("c o", "k o");
ss1 = ss1.Replace("c u", "k u");
ss1 = ss1.Replace("^ j", "^ $");
ss1 = ss1.Replace("uu", "u");
ss1 = ss1.Replace("aa", "a");
ss1 = ss1.Replace("oo", "o");
ss1 = ss1.Replace("ee", "e");
ss1 = ss1.Replace("g ", "g^");
ss1 = ss1.Replace("8", "8^");
ss1 = ss1.Replace("9", "9*");
ss1 = ss1.Replace("7", "(");
//ss1=String.TrimTrailingBlanks(String.TrimLeadingBlanks(ss1))

```

```

int ispeedneg = Convert.ToInt32(speedneg);
int ispeedplus = Convert.ToInt32(speedplus);
int speed = 50 + ispeedneg - ispeedplus;

```

```

ss1 = ss1.Replace("j", "j" + " " + (speed + 43).ToString() + "\r\n");
ss1 = ss1.Replace("h", "h" + " " + (speed+33) .ToString ()+" \r\n");
ss1 = ss1.Replace("p", "p" + " " + (speed+62) .ToString ()+" \r\n");
ss1 = ss1.Replace("t", "t" + " " + (speed+31) .ToString ()+" \r\n");
ss1 = ss1.Replace("c", "c" + " " + (speed+60) .ToString ()+" \r\n");
ss1 = ss1.Replace("k", "k" + " " + (speed+60) .ToString ()+" \r\n");
ss1 = ss1.Replace("s", "s" + " " + (speed+70) .ToString ()+" \r\n");
ss1 = ss1.Replace("x", "x" + " " + (speed+59) .ToString ()+" \r\n");
ss1 = ss1.Replace("f", "f" + " " + (speed+49) .ToString ()+" \r\n");
ss1 = ss1.Replace("b", "b" + " " + (speed+20) .ToString ()+" \r\n");
ss1 = ss1.Replace("d", "d" + " " + (speed+16) .ToString ()+" \r\n");
ss1 = ss1.Replace("g", "g" + " " + (speed+28) .ToString ()+" \r\n");
ss1 = ss1.Replace("q", "q" + " " + (speed+37) .ToString ()+" \r\n");
ss1 = ss1.Replace("z", "z" + " " + (speed+36) .ToString ()+" \r\n");
ss1 = ss1.Replace("v", "v" + " " + (speed+2) .ToString ()+" \r\n");
ss1 = ss1.Replace("r", "r" + " " + (speed-12) .ToString ()+" \r\n");
ss1 = ss1.Replace("m", "m" + " " + (speed+23) .ToString ()+" \r\n");
ss1 = ss1.Replace("n", "n" + " " + (speed+11) .ToString ()+" \r\n");
ss1 = ss1.Replace("l", "l" + " " + (speed+8) .ToString ()+" \r\n");
ss1 = ss1.Replace("y", "y" + " " + (speed+2) .ToString ()+" \r\n");
ss1 = ss1.Replace("i", "i" + " " + (speed+51) .ToString ()+" \r\n");
ss1 = ss1.Replace("u", "u" + " " + (speed+62) .ToString ()+" \r\n");
ss1 = ss1.Replace("e", "e" + " " + (speed+19) .ToString ()+" \r\n");
ss1 = ss1.Replace("o", "o" + " " + (speed+33) .ToString ()+" \r\n");
ss1 = ss1.Replace("a", "a" + " " + (speed+41) .ToString ()+" \r\n");
ss1 = ss1.Replace("ow", "ow" + " " + (speed+34) .ToString ()+" \r\n");
ss1 = ss1.Replace("", "");
ss1 = ss1.Replace("%", "a:" + " " + (speed+88) .ToString ()+" \r\n");
ss1 = ss1.Replace("@", "c:" + " " + (speed+70) .ToString ()+" \r\n");

```

```

ss1 = ss1.Replace("|", "z:" + "(speed+46) .ToString ()+" \r\n");
ss1 = ss1.Replace("w", "s:" + "(speed+61) .ToString ()+" \r\n");
ss1 = ss1.Replace("$", "j:" + "(speed+42) .ToString ()+" \r\n");
ss1 = ss1.Replace("#", "g:" + "(speed+28) .ToString ()+" \r\n");
ss1 = ss1.Replace("00", "");
ss1 = ss1.Replace("*", "n" + "0" + " \r\n");
ss1 = ss1.Replace("^", "n" + "0" + " \r\n");
ss1 = ss1.Replace("(", "n" + "0" + " \r\n");
ss1 = ss1.Replace("^", "n" + "0" + " \r\n");
ss1 = ss1.Replace("_", "_ 15 \r\n" + "n 0 \r\n");
ss1 = ss1.Replace(")", "");
ss1 = "\r\n" + "n " + "0" + " \r\n" + ss1;
System.IO.File.WriteAllText(@"c:\Amir\x.pho", ss1);

```

```

//if (wavefile == 1)
//{
// Make the Wave file xx.wav
//Runit = "c:\Amir\xer.exe database=c:\Amir\kish.
c:\Amir\play.phs /O=c:\Amir\xx.wav /FORMAT=WAV";
//System.IO.File.WriteAllText(@"c:\Amir\x.bat", Runit);
// make wave file of it

```

```

if (voice == "M")
System.Diagnostics.Process.Start(@"c:\Amir\x.bat");
if (voice == "F")
System.Diagnostics.Process.Start(@"c:\Amir\x2.bat");
System.Media.SoundPlayer simpleSound = new
System.Media.SoundPlayer(@"c:\Amir\xx.wav");
simpleSound.Play();
//}

```

```

// if (voice=="M")
// // Male Voice
//System.Diagnostics.Process.Start(@"c:\Amir\play.phs");
// if (voice=="F")
// // Female Voice
//System.Diagnostics.Process.Start(@"c:\Amir\play2.phs");
// if (voice == "B")
// {
//
//System.Diagnostics.Process.Start(@"c:\Amir\play.phs");
//
//System.Diagnostics.Process.Start(@"c:\Amir\play2.phs");
// }

```

```

private void checkBox1_CheckedChanged(object sender,
EventArgs e)
{
if (password2() == 1)
{
cancel(" " + nword);
mbrola(outit);
}
}

```

```

}
public string combination(string x)
{
string out1 = "--", out2 = "--", rest = "";
for (int i = 1; i <= x.Length - 2; i++) {
out1 = search(x.Substring(0, x.Length - i));
out2 = search(x.Substring(x.Length - i, i));
if (out1 != "--" && out2 != "--")
{
rest = out1.Substring(out1.Length - 2, 2);
out1 = out1.Substring(0, out1.Length - 2);
out2 = out2.Substring(1, out2.Length - 3);
return out1 + out2 + rest;
}
}
return "--";
}

```

```

public string priwordsearch(string x)
{
string outright1 = "--", outright2 = "--", outright3 = "--",
left1 = "--", left2 = "--", left3 = "---", right1 = "--", right2 = "--", right3 = "--";
if (x.Length < 3) x = x + "--";
left1 = x.Substring(0, 1); left2 = x.Substring(0, 2); left3 =
x.Substring(0, 3); right1 = x.Substring(1, x.Length - 1); right2 =
x.Substring(2, x.Length - 2); right3 = x.Substring(3, x.Length -
3);
if (left1 == "b" || left1 == "m" || left1 == "n" || left2 ==
"my" || left2 == "al" || left3 == "nmy")
{
outright1 = search(right1);
outright2 = search(right2);
outright3 = search(right3);
if (left1 == "m" && outright1 != "--") return "O" +
"ma" + outright1.Substring(1, outright1.Length - 1);
if (left1 == "b" && outright1 != "--") return "O" + "be"
+ outright1.Substring(1, outright1.Length - 1);
if (left1 == "n" && outright1 != "--") return "O" + "na"
+ outright1.Substring(1, outright1.Length - 1);
if (left2 == "my" && outright2 != "--") return "O" +
"mi" + outright2.Substring(1, outright2.Length - 1);
if (left2 == "al" && outright2 != "--") return "O" + "al"
+ outright2.Substring(1, outright2.Length - 1);
if (left3 == "nmy" && outright3 != "--") return "O" +
"nemi" + outright3.Substring(1, outright3.Length - 1);
}
return "--";
}

```

```

public string priwordcombination(string x)
{
string outright1 = "--", outright2 = "--", outright3 = "--",
left1 = "--", left2 = "--", left3 = "---", right1 = "--", right2 = "--",
right3 = "---";
if (x.Length < 3) x = x + "--";
left1 = x.Substring(0, 1); left2 = x.Substring(0, 2); left3 =
x.Substring(0, 3); right1 = x.Substring(1, x.Length - 1); right2 =
x.Substring(2, x.Length - 2); right3 = x.Substring(3, x.Length -
3);

```



```
//}
//}
//}
//}
//}
//GloGGrammer=GloGGrammer+" "+StringLeft(search,2)
//return
//EndFunction
```

```
public string SearchInWhat(string nnword, string nword,
string moute, string s2, string s3, int number, int lastnomre, int
lennword, string s0)
```

```
{
    string p, search;

    search = "Sax";
    if (number == lastnomre)
        search = "Vax";
    else
    {
        if (moute == "N6" || s0 == "on")
            search = "Nax";
        else
        {
            if (s2 == "ta" || s2 == "" || s2 == "v")
                search = "Vax";
            else
            {
                if (s2.Length < 3) s2 = s2 + "---";

                if (s2 == "ra" || moute == "A1" || s2 == "dr" ||
s2.Substring(0, 3) == "bvd")
                    search = "Nax";
                else
                {
                    if (s2 == ".")
                        search = "Vax";
                    else
                    {
                        if (s2.Length < 1)
                            search = "Vax";
                        else
                        {p="-----";
                            p = s2+p ;
                            if (p == "." || p.Substring(0, 7) ==
"nmygzar" || p.Substring(0, 6) == "mygzar" || p.Substring(0, 5)
== "bgzar" || p.Substring(0, 5) == "ngzar" || p.Substring(0, 4) ==
"gzar" || p.Substring(0, 3) == "krd" || p.Substring(0, 3) == "grd" ||
p.Substring(0, 4) == "nkrd" || p.Substring(0, 5) == "mykrd" ||
p.Substring(0, 2) == "kn" || p.Substring(0, 3) == "bkn" ||
p.Substring(0, 3) == "nkn" || p.Substring(0, 4) == "mykn" ||
p.Substring(0, 3) == "bzn" || p.Substring(0, 3) == "nzn" ||
p.Substring(0, 3) == "bvd" || p == "kh")
                            search = "Vax";
                        else
                        {
                            p = "-----";
                            p = s3+p ;
```

```
        if (p.Substring(0, 4) == "xvah" ||
p.Substring(0, 6) == "myxvah")
            search = "Vax";
        else
        {
            if (s2 == "bvd" || s2 == "nbvd" || s2
== "mbvd" || s2 == "nmybvd" || s2 == "bvdm" || s2 == "bvdy" ||
s2 == "bvdym" || s2 == "bvdyd" || s2 == "bvdnd")
                search = "Nax";
            else
            {
                if (moute.Substring(0, 1) == "N")
                    //;;search="Aax"
                    search = "Nax";
                else
                {
                    if (moute == "Pr" || s2 == "ra" ||
moute == "FE" || moute == "*" || moute == "CO" || moute ==
"A1")
                        search = "Nax";
                    else
                    {
                        if (moute == "Po")
                            search = "Vax";
                        else
                        {
                            if (moute == "CO" || moute
== "CO" || s0 == "on")
                                search = "Nax";
                        }
                    }
                }
            }
        }
    }
}
return search;
}
```

```
public int issamex(string xword,int len)
{
    if (xword=="stan") return 1;
    if(same(xword)==1) return 1;
    else return 0;
}
```

```
public int IsexsameX(string xword,int len)
{
    string testword,L1,R1,R2;
    if (xword.Length >0){
        L1=xword.Substring (0,1);
        R1=xword.Substring(xword.Length-1,1);
```

```

R2=xword.Substring(xword.Length-2,2);

if ( ( L1=="b" || L1=="m" || L1=="n" ) && ( R2=="ym" ||
R2=="yd" || R2=="nd" ) )
{
testword= xword .Substring (1,xword .Length -1);//
StringChopLeft(xword,1)
testword= xword .Substring (0,xword .Length -2);//
StringChopRight(testword,2)
if( issamex(testword,testword.Length) ==1) return 1;
else return 0;
}

if ( ( L1 == "b" || L1=="m" || L1=="n" ) && ( R1=="m" ||
R1=="y" || R1=="n" ) ) {
testword= xword .Substring (1,xword .Length -1);//
StringChopLeft(xword,1)
testword= xword .Substring (0,xword .Length -1);//
StringChopRight(testword,1)
if( issamex(testword,testword.Length) ==1) return 1;
else return 0;
}

if (L1=="b" || L1=="m" || L1=="n") {
testword= xword .Substring (1,xword .Length -1);//
StringChopLeft(xword,1)testword=StringChopLeft(xword,1)
if( issamex(testword,testword.Length) ==1) return 1;
else return 0;
}

if (R2=="ym" || R2=="yd" || R2=="nd" ) {
testword= xword .Substring (0,xword .Length -2);//
StringChopRight(testword,2)testword=StringChopRight(xword,2
)
if( issamex(testword,testword.Length) ==1) return 1;
else return 0;
}

if (R1 == "m" || R1 == "y" || R1 == "n")
{
testword = xword.Substring(0, xword.Length - 1);//
StringChopRight(testword,2)testword=StringChopRight(xword,1
)
if( issamex(testword, testword.Length) == 1) return 1;
else return 0;
}
}

return 0;
}

public int same(string xword)
{ //a[438] to a[669]

for (int i = 438; i <= 669; i++)
if (a[i] == xword) return 1;
return 0;
}

public string noun(string xword)
{ //a[1] to a[437]

```

```

string result = "--";
for (int i = 1; i <= 437; i++)
{
if (a[i] == xword)
{
result = b[i];
}
else
{
}
}
}
return result;
}

public string verb(string xword)
{ //a[438] to a[669]
string result = "--";
for (int i = 438; i <= 669; i++)
{
if (a[i] == xword)
{
result = b[i];
}
else
{
}
}
}
return result;
}

private void radioButton2_CheckedChanged(object sender,
EventArgs e)
{
if (radioButton2.Checked)
voice = "F";
}

public string Search3Number(string tword)
{

string strtt;
string eqnum;
string hez;

string s1;
string sense;
string sad;
string dah;
string mnum;
string ssamir;
string sen;
string cheq;
int ssamirv;

```

```

string z;
int zv;
string sx;
string zz;
int zzv;
int lsen;
string glonum="--";
string dahh;
string yekh;
string sadh;
//totalnumber="";
s1 = tword;
strtt = "";
eqnum = "";
hez = "";
sad = "";
dah = "";
rnum = "";
ssamir = tword.Replace(" ", "");
ssamir = ssamir.Replace(" ", ""); ;//.Replace("O", "");
if (ssamir.Length >0)
{
    if (ssamir.Substring(0, 1) == "0")
        ssamir = ssamir.Substring(1, ssamir.Length - 1);
}
if (ssamir.Length > 0)
{
    if (ssamir.Substring(0, 1) == "0")
        ssamir = ssamir.Substring(1, ssamir.Length - 1);
}
if (ssamir.Length > 0)
{
    if (ssamir.Substring(0, 1) == "0")
        ssamir = ssamir.Substring(1, ssamir.Length - 1);
}

sen = ssamir;
cheq = "";
if (sen.Length > 0)
{
    if (sen.Length > 1)
        cheq = sen.Substring(sen.Length - 2, 2);
//StringRight(sen,2)

    // sense = sen;
    lsen = sen.Length; ;//StringLength(sen)

    if (lsen == 1)
    {
        // dah = "";
        // yekh = "";
        // dahh = "";
        // sadh = "";
        // sad = "";
        rnum = "";
        //ssamirv = Convert.ToInt32(ssamir); //
StringToInt(ssamir)

        if (ssamir == "0")
            rnum = "sefr";

        if (ssamir == "1")
            rnum = "yek";

        if (ssamir == "2")
            rnum = "do";

        if (ssamir == "3")
            rnum = "se";

        if (ssamir == "4")
            rnum = "c1h1r";

        if (ssamir == "5")
            rnum = "panj";

        if (ssamir == "6")
            rnum = "wew";

        if (ssamir == "7")
            rnum = "haft";

        if (ssamir == "8")
            rnum = "hawt";

        if (ssamir == "9")
            rnum = "noh";

        return rnum;
    }

    if (lsen == 2)
    {
        if (lsen == 2 && Convert.ToInt32(cheq) < 20)//
StringToInt(cheq)<20)
        {
            dah = "";
            yekh = "";
            dahh = "";
            sadh = "";
            sad = "";
            rnum = "";
            ssamirv = Convert.ToInt32(ssamir);
            //StringToInt(ssamir)
            if (ssamirv == 10)
                rnum = "dah";

            if (ssamirv == 11)
                rnum = "y1zdah";

            if (ssamirv == 12)
                rnum = "dav1zah";

            if (ssamirv == 13)
                rnum = "s1zdah";

            if (ssamirv == 14)
                rnum = "c1h1rdah";
        }
    }
}

```

```

if (ssamirv == 15)
    rnum = "p1nzdah";

if (ssamirv == 16)
    rnum = "w1nzdah";

if (ssamirv == 17)
    rnum = "hefdah";

if (ssamirv == 18)
    rnum = "hejdah";

if (ssamirv == 19)
    rnum = "nuzdah";

return rnum;
}

if (lсен == 2 && Convert.ToInt32(cheq) > 19)
//StringToInt(cheq)>19)
{
    dah = "";
    yekh = "";
    dahh = "";
    sadh = "";
    sad = "";
    rnum = "";
    z = сен.Substring(0, 1); //StringLeft(сен,1)
    ssamir = сен.Substring(сен.Length - 1, 1);
//StringRight(сен,1)
    ssamirv = Convert.ToInt32(ssamir);
//StringToInt(ssamir)
    zv = Convert.ToInt32(z); //StringToInt(z)
    if (ssamirv == 1)
        rnum = "o" + "yek";

    if (ssamirv == 2)
        rnum = "o" + "do";

    if (ssamirv == 3)
        rnum = "o" + "se";

    if (ssamirv == 4)
        rnum = "o" + "clh1r";

    if (ssamirv == 5)
        rnum = "o" + "panj";

    if (ssamirv == 6)
        rnum = "o" + "wew";

    if (ssamirv == 7)
        rnum = "o" + "haft";

    if (ssamirv == 8)
        rnum = "o" + "hawt";

    if (ssamirv == 9)
        rnum = "o" + "noh";

    if (zv == 2)
        dah = "bist";

    if (zv == 3)
        dah = "si";

    if (zv == 4)
        dah = "cehel";

    if (zv == 5)
        dah = "panj1h";

    if (zv == 6)
        dah = "3ast";

    if (zv == 7)
        dah = "haft1d";

    if (zv == 8)
        dah = "hawt1d";

    if (zv == 9)
        dah = "navad";

    return dah + rnum;
}

//;.....; 3 3 3 3
3 .....

if (lсен == 3)
{
    dah = "";
    sad = "";
    rnum = "";
    cheq = сен.Substring(сен.Length - 2, 2); //
StringRight(сен,2)
    if (Convert.ToInt32(cheq) < 10 ||
Convert.ToInt32(cheq) > 19)
        { //StringToInt(cheq)<10 || StringToInt(cheq)>19 )
            ssamir = сен.Substring(сен.Length - 1, 1);
//StringRight(сен,1)
            sx = сен.Substring(0, 2); //StringLeft(сен,2)
            z = сен.Substring(0, 1); //StringLeft(сен,1)
            zz = sx.Substring(sx.Length - 1, 1);
//StringRight(sx,1)
            ssamirv = Convert.ToInt32(ssamir);
//StringToInt(ssamir)
            zv = Convert.ToInt32(z); //StringToInt(z)
            zzv = Convert.ToInt32(zz); //StringToInt(zz)
            if (ssamirv == 1)
                rnum = "o" + "yek";

            if (ssamirv == 2)
                rnum = "o" + "do";
        }
}

```



```

if (ssamirv == 3)
    rnum = "o" + "se";
if (ssamirv == 4)
    rnum = "o" + "clh1r";
if (ssamirv == 5)
    rnum = "o" + "panj";
if (ssamirv == 6)
    rnum = "o" + "wew";
if (ssamirv == 7)
    rnum = "o" + "haft";
if (ssamirv == 8)
    rnum = "o" + "hawt";
if (ssamirv == 9)
    rnum = "o" + "noh";
if (zzv == 2)
    dah = "o" + "bist";
if (zzv == 3)
    dah = "o" + "si";
if (zzv == 4)
    dah = "o" + "cehel";
if (zzv == 5)
    dah = "o" + "panj1h";
if (zzv == 6)
    dah = "o" + "wast";
if (zzv == 7)
    dah = "o" + "haft1d";
if (zzv == 8)
    dah = "o" + "hawt1d";
if (zzv == 9)
    dah = "o" + "navad";
if (zv == 2)
    sad = "divist";
if (zv == 3)
    sad = "sisad";
if (zv == 5)
    sad = "p1nsad";
if (zv == 1)
    sad = "sad";
if (zv == 4)
    sad = "clh1r" + "sad";
if (zv == 6)
    sad = "wew" + "sad";
if (zv == 7)
    sad = "haft" + "sad";
if (zv == 8)
    sad = "hawt" + "sad";
if (zv == 9)
    sad = "noh" + "sad";
return sad + dah + rnum;
}

if (Convert.ToInt32(cheq) < 20 &&
Convert.ToInt32(cheq) > 9)
    { //StringToInt(cheq)<20 && StringToInt(cheq)>9 )
        dah = "";
        sad = "";
        rnum = "";
        ssamir = sen.Substring(sen.Length - 2, 2);
        //StringRight(sen,2)
        z = sen.Substring(0, 1); //StringLeft(sen,1)
        ssamirv = Convert.ToInt32(ssamir);
        //StringToInt(ssamir)
        zv = Convert.ToInt32(z); //StringToInt(z)
        if (ssamirv == 10)
            rnum = "o" + "dah";
        if (ssamirv == 11)
            rnum = "o" + "y1zdah";
        if (ssamirv == 12)
            rnum = "o" + "dav1zah";
        if (ssamirv == 13)
            rnum = "o" + "sizardah";
        if (ssamirv == 14)
            rnum = "o" + "clh1rdah";
        if (ssamirv == 15)
            rnum = "o" + "p1nzdah";
        if (ssamirv == 16)
            rnum = "o" + "w1nzdah";
        if (ssamirv == 17)
            rnum = "o" + "hefdah";
        if (ssamirv == 18)
            rnum = "o" + "hejdah";
        if (ssamirv == 19)
            rnum = "o" + "nuzdah";
        if (zv == 2)
            sad = "divist";
        if (zv == 3)
    
```

