Iniciació a la Criptografia

Artur Travesa

(versió 2024-07)

Capítol 0. Codificació

0.0. Introducció

La idea bàsica que guia aquest primer capítol de les notes és descriure l'aplicació d'eines d'Aritmètica bàsica que són fonamentals per a les transmissions d'informació, xifrades o no.

I per a la transmissió d'informació, cal tenir en compte, en primer lloc, quin tipus de missatges es volen transmetre i com es poden transcriure de manera que se'n pugui fer un estudi i una manipulació raonablement còmodes; és a dir, com estan *codificats*.

Centrem, doncs, el contingut, en la descripció de la codificació bàsica que considerarem per als missatges.

0.1. Codificació de missatges

Sembla evident que la codificació dels missatges dependrà en gran manera de quin tipus de missatges es volen transmetre. Per exemple, és molt difícil pensar com un actor pot transmetre una olor, o una sensació tàctil, o un gust, o una emoció qualsevol, almenys, de manera exacta o, fins i tot, prou acurada, a algun altre.

Fins i tot, un missatge tan senzill (de dir) com "t'estimo", no es pot transmetre amb tota la seva profunditat sense la presència en directe de l'emissor i el receptor del missatge; en efecte, no només és la frase concreta, sinó la mirada, el to, el timbre de veu, i el context, per posar només un exemple, allò que fa que el missatge arribi completament al seu destinatari. I, fins i tot així, potser el missatge rebut no és exactament el missatge enviat! La "codificació" del missatge no és prou precisa perquè el missatge pugui arribar íntegrament al seu destinatari, tal com el vol transmetre l'emissor.

Observació. Per als missatges criptogràfics, això no és possible. En efecte, **CAL** que el missatge desxifrat sigui exactament igual que el missatge que es xifra; per tant, el codificador i el descodificador hauran de ser adequats a fi de fer això possible. En particular, això limitarà el tipus de missatge que es pot xifrar/desxifrar.

Doncs, sembla evident que cal que l'emissor disposi d'un "codificador" adaptat al tipus de missatge que vol transmetre, i que el destinatari disposi d'un "descodificador" adequat que recuperi el missatge enviat, si és que això és possible.

Per exemple, per a la transmissió oral d'informació, cal que l'emissor i el receptor disposin d'una llengua comuna per a la transmissió del missatge; en aquest cas, la llengua actua com a codificador i també com a descodificador del missatge. Notem, també, que, tot i així, el procés de codificació i descodificació no sempre aconsegueix una transmissió perfecta del missatge.

Però encara que els dos actors disposin d'aquesta llengua comuna, la transmissió del missatge no és xifrada; només és codificada. De fet, qualsevol altre actor que conegui aquesta llengua comuna podria descodificar el missatge i aquest no li seria críptic. És a dir, qualsevol actor que conegui el sistema de codificació del missatge el pot descodificar sense conèixer cap altra informació complementària. En aquest cas, no parlem de Criptografia, sinó només de Codificació.

I abans de fer críptics els missatges per a altres actors, cal que el missatge sigui codificat d'alguna manera adequada a la seva transmissió.

Per a començar un estudi matemàtic de la criptografia, cal concretar bé quin tipus de missatges podem tractar, a fi de poder-los codificar de manera adequada i, després, fer-los críptics a tercers actors.

Un dels fets bàsics que s'han de tenir presents, és que la codificació pot dependre molt del tipus de missatge que es vol transmetre. Per exemple, potser cal transmetre dades bancàries per a una transacció comercial (NIF, números de comptes corrents, números de targetes de crèdit, quantitats que cal transferir entre comptes diferents, etcètera), o bé cal transmetre informacions confidencials alfanumèriques en general (informes de seguiment de projectes, cartes de recomanació, sol·licituds de beques), o bé cal transmetre claus privades per a l'ús de certes funcions criptogràfiques (per exemple, en les comunicacions xifrades entre diferents ordinadors), etcètera.

Observació. Això obvia, clarament, quin és el procés bàsic de codificació al qual s'han de sotmetre prèviament els missatges. Per exemple, un text alfanumèric es codifica, sovint, amb un processador de textos concret, que li dóna una forma tal que qualsevol altre actor amb el mateix processador de textos el pot recuperar íntegrament; però potser no el pot recuperar amb un processador diferent. És conegut que per a evitar aquest tipus de problemes "d'incompatibilitat" entre els sistemes diferents, es poden fer servir formats de tipus estàndard. I, encara ens podem trobar amb més dificultats si volem tractar altres tipus de missatges; per exemple, de so, o d'imatge; o bé d'arxius executables en un determinat sistema operatiu.

0.1.0. Tipus de missatges

Actualment, la majoria de les comunicacions s'estableixen entre ordinadors, i les dades que cal transmetre són, sovint, arxius (de text, o executables, o multimèdia comprimits, o del tipus que sigui). Per tant, volem suposar, i ho farem sovint, que les dades que cal transmetre són successions de zeros i uns.

A fi de poder *simular* còmodament aquest fet, en aquest tutorial usarem les funcions incorporades chr( ) i ord( ). Cal notar que *SageMath* usa els 1114112 codis numèrics Unicode, entre els valors 0 i $16^5 + 16^4 - 1 = 1114111$=0x10ffff, inclosos; o sigui en el

rang (en el tipus de dades de *SageMath*) $0 \leq codi < 1114112$.

Notem que es tracta només d'una *simulació*. En un cas real, caldria treballar directament sobre els bits (o, potser, sobre els bytes) de l'arxiu origen. En efecte, els diferents programaris *interpreten* d'alguna manera les successions de bits que constitueixen l'arxiu i ens el presenten visualment; però usualment no ens ensenyen la successió original de bits. De fet, nosaltres treballarem sobre aquesta *interpretació*.

0.2. La llista dels caràcters de *SageMath*: de codis a caràcters

0.2.0. Els caràcters ASCII

Els caràcters ASCII (American Standard Code for Information Interchange) imprimibles són els caràcters de codis entre el 32 (caràcter blanc=espai) i el 126 (titlle). Fem-ne una llista.

In [1]:
```
print([[i,chr(i)] for i in range(32,127)])
```

```
[[32, ' '], [33, '!'], [34, '"'], [35, '#'], [36, '$'], [37, '%'],
[38, '&'], [39, "'"], [40, '('], [41, ')'], [42, '*'], [43, '+'],
[44, ','], [45, '-'], [46, '.'], [47, '/'], [48, '0'], [49, '1'],
[50, '2'], [51, '3'], [52, '4'], [53, '5'], [54, '6'], [55, '7'],
[56, '8'], [57, '9'], [58, ':'], [59, ';'], [60, '<'], [61, '='],
[62, '>'], [63, '?'], [64, '@'], [65, 'A'], [66, 'B'], [67, 'C'],
[68, 'D'], [69, 'E'], [70, 'F'], [71, 'G'], [72, 'H'], [73, 'I'],
[74, 'J'], [75, 'K'], [76, 'L'], [77, 'M'], [78, 'N'], [79, 'O'],
[80, 'P'], [81, 'Q'], [82, 'R'], [83, 'S'], [84, 'T'], [85, 'U'],
[86, 'V'], [87, 'W'], [88, 'X'], [89, 'Y'], [90, 'Z'], [91, '['],
[92, '\\'], [93, ']'], [94, '^'], [95, '_'], [96, '`'], [97, 'a'],
[98, 'b'], [99, 'c'], [100, 'd'], [101, 'e'], [102, 'f'], [103,
'g'], [104, 'h'], [105, 'i'], [106, 'j'], [107, 'k'], [108, 'l'],
[109, 'm'], [110, 'n'], [111, 'o'], [112, 'p'], [113, 'q'], [114,
'r'], [115, 's'], [116, 't'], [117, 'u'], [118, 'v'], [119, 'w'],
[120, 'x'], [121, 'y'], [122, 'z'], [123, '{'], [124, '|'], [125,
'}'], [126, '~']]
```

Abans dels caràcters ASCII imprimibles (del 0 al 31), i després (només el 127), hi ha els diferents caràcters ASCII de control; mirem-nos-els, i notem que *SageMath* ens en dóna una interpretació en format hexadecimal d'un sol byte, excepte dels codis *tabulador*, \t, *línia nova*, \n, i *retorn de carro*, \r.

In [2]:
```
print([[i,chr(i)] for i in range(0,32)]+[[127,chr(127)]])
```

```
[[0, '\x00'], [1, '\x01'], [2, '\x02'], [3, '\x03'], [4, '\x04'],
[5, '\x05'], [6, '\x06'], [7, '\x07'], [8, '\x08'], [9, '\t'], [10,
'\n'], [11, '\x0b'], [12, '\x0c'], [13, '\r'], [14, '\x0e'], [15,
'\x0f'], [16, '\x10'], [17, '\x11'], [18, '\x12'], [19, '\x13'], [2
0, '\x14'], [21, '\x15'], [22, '\x16'], [23, '\x17'], [24, '\x18'],
[25, '\x19'], [26, '\x1a'], [27, '\x1b'], [28, '\x1c'], [29, '\x1
d'], [30, '\x1e'], [31, '\x1f'], [127, '\x7f']]
```

0.2.1. Extensions del codi ASCII

Després dels caràcters ASCII imprimibles hi ha diferents extensions de la taula de codis fins al codi 255 (o més enllà). Són les *diferents taules de caràcters* que corresponen a les diferents extensions del codi ASCII que s'han usat durant molt de temps; de fet, n'hi ha moltes i convé utilitzar-ne una més estandarditzada. *SageMath* utilitza la taula de codis estàndard *Unicode*. Notem que per als primers caràcters que interpreta però no "dibuixa" en proporciona (com abans) una codificació hexadecimal d'un sol byte (notem que $255 = 2^8 - 1 = 16^2 - 1$ s'escriu, en base 16, com "ff" i, en binari, com "1111 1111").

In [3]:
```
print([chr(v) for v in range(128,256)])
```

```
['\x80', '\x81', '\x82', '\x83', '\x84', '\x85', '\x86', '\x87',
'\x88', '\x89', '\x8a', '\x8b', '\x8c', '\x8d', '\x8e', '\x8f', '\x
90', '\x91', '\x92', '\x93', '\x94', '\x95', '\x96', '\x97', '\x9
8', '\x99', '\x9a', '\x9b', '\x9c', '\x9d', '\x9e', '\x9f', '\xa0',
'¡', '¢', '£', '¤', '¥', '¦', '§', '¨', '©', 'ª', '«', '¬', '\xad',
'®', '¯', '°', '±', '²', '³', '´', 'µ', '¶', '·', '¸', '¹', 'º',
'»', '¼', '½', '¾', '¿', 'À', 'Á', 'Â', 'Ã', 'Ä', 'Å', 'Æ', 'Ç',
'È', 'É', 'Ê', 'Ë', 'Ì', 'Í', 'Î', 'Ï', 'Ð', 'Ñ', 'Ò', 'Ó', 'Ô',
'Õ', 'Ö', '×', 'Ø', 'Ù', 'Ú', 'Û', 'Ü', 'Ý', 'Þ', 'ß', 'à', 'á',
'â', 'ã', 'ä', 'å', 'æ', 'ç', 'è', 'é', 'ê', 'ë', 'ì', 'í', 'î',
'ï', 'ð', 'ñ', 'ò', 'ó', 'ô', 'õ', 'ö', '÷', 'ø', 'ù', 'ú', 'û',
'ü', 'ý', 'þ', 'ÿ']
```

In [4]:
```
print([chr(v) for v in range(256,512)])
```

```
['Ā', 'ā', 'Ă', 'ă', 'Ą', 'ą', 'Ć', 'ć', 'Ĉ', 'ĉ', 'Ċ', 'ċ', 'Č',
'č', 'Ď', 'ď', 'Đ', 'đ', 'Ē', 'ē', 'Ĕ', 'ĕ', 'Ė', 'ė', 'Ę', 'ę',
'Ě', 'ě', 'Ĝ', 'ĝ', 'Ğ', 'ğ', 'Ġ', 'ġ', 'Ģ', 'ģ', 'Ĥ', 'ĥ', 'Ħ',
'ħ', 'Ĩ', 'ĩ', 'Ī', 'ī', 'Ĭ', 'ĭ', 'Į', 'į', 'İ', 'ı', 'Ĳ', 'ĳ',
'Ĵ', 'ĵ', 'Ķ', 'ķ', 'ĸ', 'Ĺ', 'ĺ', 'Ļ', 'ļ', 'Ľ', 'ľ', 'Ŀ', 'ŀ',
'Ł', 'ł', 'Ń', 'ń', 'Ņ', 'ņ', 'Ň', 'ň', 'ŉ', 'Ŋ', 'ŋ', 'Ō', 'ō',
'Ŏ', 'ŏ', 'Ő', 'ő', 'Œ', 'œ', 'Ŕ', 'ŕ', 'Ŗ', 'ŗ', 'Ř', 'ř', 'Ś',
'ś', 'Ŝ', 'ŝ', 'Ş', 'ş', 'Š', 'š', 'Ţ', 'ţ', 'Ť', 'ť', 'Ŧ', 'ŧ',
'Ũ', 'ũ', 'Ū', 'ū', 'Ŭ', 'ŭ', 'Ů', 'ů', 'Ű', 'ű', 'Ų', 'ų', 'Ŵ',
'ŵ', 'Ŷ', 'ŷ', 'Ÿ', 'Ź', 'ź', 'Ż', 'ż', 'Ž', 'ž', 'ſ', 'ƀ', 'Ɓ',
'Ƃ', 'ƃ', 'Ƅ', 'ƅ', 'Ɔ', 'Ƈ', 'ƈ', 'Ɖ', 'Ɗ', 'Ƌ', 'ƌ', 'ƍ', 'Ǝ',
'Ə', 'Ɛ', 'Ƒ', 'ƒ', 'Ɠ', 'Ɣ', 'ƕ', 'Ɩ', 'Ɨ', 'Ƙ', 'ƙ', 'ƚ', 'ƛ',
'Ɯ', 'Ɲ', 'ƞ', 'Ɵ', 'Ơ', 'ơ', 'Ƣ', 'ƣ', 'Ƥ', 'ƥ', 'Ʀ', 'Ƨ', 'ƨ',
'Σ', 'Ʃ', 'ƪ', 'Ƭ', 'ƭ', 'Ʈ', 'Ư', 'ư', 'Ʊ', 'Ʋ', 'Ƴ', 'ƴ', 'Ƶ',
'ƶ', 'Ʒ', 'Ƹ', 'ƹ', 'ƺ', 'Ƽ', 'ƽ', 'ƾ', 'ƿ', 'ǀ', 'ǁ', 'ǂ', 'ǃ',
'DŽ', 'Dž', 'dž', 'LJ', 'Lj', 'lj', 'NJ', 'Nj', 'nj', 'Ǎ', 'ǎ',
'Ǐ', 'ǐ', 'Ǒ', 'ǒ', 'Ǔ', 'ǔ', 'Ǖ', 'ǖ', 'Ǘ', 'ǘ', 'Ǚ', 'ǚ', 'Ǜ',
'ǜ', 'ǝ', 'Ǟ', 'ǟ', 'Ǡ', 'ǡ', 'Ǣ', 'ǣ', 'Ǥ', 'ǥ', 'Ǧ', 'ǧ', 'Ǩ',
'ǩ', 'Ǫ', 'ǫ', 'Ǭ', 'ǭ', 'Ǯ', 'ǯ', 'ǰ', 'DZ', 'Dz', 'dz', 'Ǵ',
'ǵ', 'Ƕ', 'Ƿ', 'Ǹ', 'ǹ', 'Ǻ', 'ǻ', 'Ǽ', 'ǽ', 'Ǿ', 'ǿ']
```

In [5]:
```
1 print([chr(v) for v in range(512,1024)])
```

```
['À', 'à', 'Â', 'â', 'È', 'è', 'Ê', 'ê', 'Ì', 'ì', 'Î', 'î', 'Ò',
'ò', 'Ô', 'ô', 'Ř', 'ř', 'Ŕ', 'ŕ', 'Ù', 'ù', 'Û', 'û', 'Ş', 'ş',
'Ţ', 'ţ', 'Ȝ', 'ȝ', 'Ȟ', 'ȟ', 'ȵ', 'ȶ', 'Ȣ', 'ȣ', 'Ȥ', 'ȥ', 'Ȧ',
'ȧ', 'Ȩ', 'ȩ', 'Ȫ', 'ȫ', 'Ȭ', 'ȭ', 'Ȯ', 'ȯ', 'Ȱ', 'ȱ', 'Ȳ', 'ȳ',
'ȴ', 'ȶ', 'ȶ', 'ȷ', 'ȸ', 'ȹ', 'Ⱥ', 'Ȼ', 'ȼ', 'Ⱦ', 'ȿ', 'ɀ', 'Ɂ',
'ɂ', 'Ƀ', 'Ʉ', 'Ʌ', 'Ɇ', 'ɇ', 'Ɉ', 'ɉ', 'Ɋ', 'ɋ', 'Ɍ', 'ɍ', 'Ɏ',
'ɏ', 'ɐ', 'ɑ', 'ɒ', 'ɓ', 'ɔ', 'ɕ', 'ɖ', 'ɗ', 'ɘ', 'ə', 'ɚ',
'ɛ', 'ɜ', 'ɝ', 'ɞ', 'ɟ', 'ɠ', 'ɡ', 'ɢ', 'ɣ', 'ɤ', 'ɥ', 'ɦ', 'ɧ',
'ɨ', 'ɩ', 'ɪ', 'ɫ', 'ɬ', 'ɭ', 'ɯ', 'ɰ', 'ɱ', 'ɲ', 'ɳ', 'ɴ',
'ɵ', 'ɶ', 'ɷ', 'ɸ', 'ɹ', 'ɺ', 'ɻ', 'ɼ', 'ɽ', 'ɾ', 'ɿ', 'ʀ', 'ʁ',
'ʂ', 'ʃ', 'ʄ', 'ʅ', 'ʆ', 'ʇ', 'ʈ', 'ʉ', 'ʊ', 'ʋ', 'ʌ', 'ʍ', 'ʎ',
'ʏ', 'ʐ', 'ʑ', 'ʒ', 'ʓ', 'ʔ', 'ʕ', 'ʖ', 'ʗ', 'ʘ', 'ʙ', 'ʚ', 'ʛ',
'ʜ', 'ʝ', 'ʞ', 'ʟ', 'ʠ', 'ʡ', 'ʢ', 'ʣ', 'ʤ', 'ʥ', 'ʦ', 'ʧ', 'ʨ',
'ʩ', 'ʪ', 'ʫ', 'ʬ', 'ʭ', 'ʮ', 'ʯ', 'ʰ', 'ʱ', 'ʲ', 'ʳ', 'ʴ', 'ʵ',
'ʶ', 'ʷ', 'ʸ', 'ʹ', 'ʺ', 'ʻ', 'ʼ', 'ʽ', 'ʾ', 'ʿ', 'ˀ', 'ˁ', '˂',
'˃', '˄', '˅', '˄', '˅', '˃', '˗', '˂', '˃', '˄', '˅', '˄', '˅',
'˒', '˓', '˔', '˕', '˖', '˗', '˘', '˙', '˚', '˛', '˜', '˝',
'˞', '˟', '˃', '˂', '˥', '˦', '˧', '˨', '˩', '˪', '˫', 'ˬ', '˭',
'ˮ', '˯', '˰', '˱', '˲', '˳', '˴', '˵', '˶', '˷', '˸', '˹', '˺',
'˻', '˼', '˽', '˾', '˿', '̀', '́', '̂', '̃', '̄', '̅', '̆', '̇',
'̈', '̉', '̊', '̋', '̌', '̍', '̎', '̏', '̐', '̑', '̒', '̓', '̔',
'̕', '̖', '̗', '̘', '̙', '̚', '̛', '̜', '̝', '̞', '̟', '̠', '̡',
'̢', '̣', '̤', '̥', '̦', '̧', '̨', '̩', '̪', '̫', '̬', '̭', '̮',
'̯', '̰', '̱', '̲', '̳', '̴', '̵', '̶', '̷', '̸', '̹', '̺', '̻',
'̼', '̽', '̾', '̿', '̀', '́', '͂', '̓', '̈́', 'ͅ', '͆', '͇', '͈',
'͉', '͊', '͋', '͌', '͍', '͎', '͏', '͐', '͑', '͒', '͓', '͔', '͕',
'͖', '͗', '͘', '͙', '͚', '͛', '͜', '͝', '͞', '͟', '͠', '͡', '͢',
'ͣ', 'ͤ', 'ͥ', 'ͦ', 'ͧ', 'ͨ', 'ͩ', 'ͪ', 'ͫ', 'ͬ', 'ͭ', 'ͮ', 'ͯ',
'Ͱ', 'ͱ', 'Ͳ', 'ͳ', 'ʹ', '͵', 'Ͷ', 'ͷ', '͸', '͹', 'ͺ', 'ͻ', 'ͼ',
'ͽ', ';', 'Ϳ', '\u0380',
'\u0381', '\u0382', '\u0383', '΄', '΅', 'Ά', '·', 'Έ', 'Ή', 'Ί',
'\u038b', 'Ό', '\u038d', 'Ύ', 'Ώ', 'ΐ', 'Α', 'Β', 'Γ', 'Δ', 'Ε',
'Ζ', 'Η', 'Θ', 'Ι', 'Κ', 'Λ', 'Μ', 'Ν', 'Ξ', 'Ο', 'Π', 'Ρ', '\u03a
2', 'Σ', 'Τ', 'Υ', 'Φ', 'Χ', 'Ψ', 'Ω', 'Ϊ', 'Ϋ', 'ά', 'έ', 'ή',
'ί', 'ΰ', 'α', 'β', 'γ', 'δ', 'ε', 'ζ', 'η', 'θ', 'ι', 'κ', 'λ',
'μ', 'ν', 'ξ', 'ο', 'π', 'ρ', 'ς', 'σ', 'τ', 'υ', 'φ', 'χ', 'ψ',
'ω', 'ϊ', 'ϋ', 'ό', 'ύ', 'ώ', 'Ϗ', 'ϐ', 'ϑ', 'ϒ', 'ϓ', 'ϔ', 'ϕ',
'ϖ', 'ϗ', 'Ϙ', 'ϙ', 'Ϛ', 'ϛ', 'Ϝ', 'ϝ', 'Ϟ', 'ϟ', 'Ϡ', 'ϡ', 'Ϣ',
'ϣ', 'Ϥ', 'ϥ', 'Ϧ', 'ϧ', 'Ϩ', 'ϩ', 'Ϫ', 'ϫ', 'Ϭ', 'ϭ', 'Ϯ', 'ϯ',
'ϰ', 'ϱ', 'ϲ', 'ϳ', 'ϴ', 'ϵ', '϶', 'Ϸ', 'ϸ', 'Ϲ', 'Ϻ', 'ϻ', 'ϼ',
'Ͻ', 'Ͼ', 'Ͽ']
```

Aquí també hi ha caràcters "interpretats", però la interpretació ja no és hexadecimal d'un sol byte, com abans, sinó que és en format unicode de 4 caràcters hexadecimals (per exemple, \u038b).

In [6]:
```
1 print([chr(v) for v in range(1024,1556)])
```

```
['Ѐ', 'Ё', 'Ђ', 'Ѓ', 'Є', 'Ѕ', 'І', 'Ї', 'Ј', 'Љ', 'Њ', 'Ћ', 'Ќ',
'Ѝ', 'Ў', 'Џ', 'А', 'Б', 'В', 'Г', 'Д', 'Е', 'Ж', 'З', 'И', 'Й',
'К', 'Л', 'М', 'Н', 'О', 'П', 'Р', 'С', 'Т', 'У', 'Ф', 'Х', 'Ц',
'Ч', 'Ш', 'Щ', 'Ъ', 'Ы', 'Ь', 'Э', 'Ю', 'Я', 'а', 'б', 'в', 'г',
'д', 'е', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р',
'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э',
'ю', 'я', 'ѐ', 'ё', 'ђ', 'ѓ', 'є', 's', 'і', 'ї', 'ј', 'љ', 'њ',
'ћ', 'ќ', 'ѝ', 'ў', 'џ', 'Ѡ', 'ѡ', 'Ѣ', 'ѣ', 'Ѥ', 'ѥ', 'Ѧ', 'ѧ',
'Ѩ', 'ѩ', 'Ѫ', 'ѫ', 'Ѭ', 'ѭ', 'Ѯ', 'ѯ', 'Ѱ', 'ѱ', 'Ѳ', 'ѳ',
'Ѵ', 'ѵ', 'Ѷ', 'ѷ', 'Ѹ', 'ѹ', 'Ѻ', 'ѻ', 'Ѽ', 'ѽ', 'Ѿ', 'ѿ',
'Ҁ', 'ҁ', '҂', '҃', '҄', '҅', '҆', '҇', '҈', '҉', 'Ҋ', 'ҋ', 'Ҍ',
'ҍ', 'Ҏ', 'ҏ', 'Ґ', 'ґ', 'Ғ', 'ғ', 'Ҕ', 'ҕ', 'Җ', 'җ', 'Ҙ', 'ҙ',
'Қ', 'қ', 'Ҝ', 'ҝ', 'Ҟ', 'ҟ', 'Ҡ', 'ҡ', 'Ң', 'ң', 'Ҥ', 'ҥ', 'Ҧ',
'ҧ', 'Ҩ', 'ҩ', 'Ҫ', 'ҫ', 'Ҭ', 'ҭ', 'Ү', 'ү', 'Ұ', 'ұ', 'Ҳ', 'ҳ',
'Ҵ', 'ҵ', 'Ҷ', 'ҷ', 'Ҹ', 'ҹ', 'Һ', 'һ', 'Ҽ', 'ҽ', 'Ҿ', 'ҿ', 'Ӏ',
'Ӂ', 'ӂ', 'Ӄ', 'ӄ', 'Ӆ', 'ӆ', 'Ӈ', 'ӈ', 'Ӊ', 'ӊ', 'Ӌ', 'ӌ', 'Ӎ',
'ӎ', 'ӏ', 'Ӑ', 'ӑ', 'Ӓ', 'ӓ', 'Ӕ', 'ӕ', 'Ә', 'ә', 'Ӛ', 'ӛ', 'Ӝ',
'ӝ', 'Ӟ', 'ӟ', 'Ӡ', 'ӡ', 'Ӣ', 'ӣ', 'Ӥ', 'ӥ', 'Ӧ', 'ӧ', 'Ө', 'ө',
'Ӫ', 'ӫ', 'Ӭ', 'ӭ', 'Ӯ', 'ӯ', 'Ӱ', 'ӱ', 'Ӳ', 'ӳ', 'Ӵ', 'ӵ', 'Ӷ',
'ӷ', 'Ӹ', 'ӹ', 'Ӻ', 'ӻ', 'Ӽ', 'ӽ', 'Ӿ', 'ӿ', 'Ԁ', 'ԁ', 'Ԃ', 'ԃ',
'Ԅ', 'ԅ', 'Ԇ', 'ԇ', 'Ԉ', 'ԉ', 'Ԋ', 'ԋ', 'Ԍ', 'ԍ', 'Ԏ', 'ԏ', 'Ԑ',
'ԑ', 'Ԓ', 'ԓ', 'Ԕ', 'ԕ', 'Ԗ', 'ԗ', 'Ԙ', 'ԙ', 'Ԛ', 'ԛ', 'Ԝ', 'ԝ',
'Ԟ', 'ԟ', 'Ԡ', 'ԡ', 'Ԣ', 'ԣ', 'Ԥ', 'ԥ', 'Ԧ', 'ԧ', 'Ԩ', 'ԩ', 'Ԫ',
'ԫ', 'Ԭ', 'ԭ', 'Ԯ', 'ԯ', '\u0530', 'Ա', 'Բ', 'Գ', 'Դ', 'Ե', 'Զ',
'Է', 'Ը', 'Թ', 'Ժ', 'Ի', 'Լ', 'Խ', 'Ծ', 'Կ', 'Հ', 'Ձ', 'Ղ', 'Ճ',
'Մ', 'Յ', 'Ն', 'Շ', 'Ո', 'Չ', 'Պ', 'Ջ', 'Ռ', 'Ս', 'Վ', 'Տ', 'Ր',
'Ց', 'Ւ', 'Փ', 'Ք', 'Օ', 'Ֆ', '\u0557', '\u0558', '֘', '֙', '֚',
'֛', '֜', '֝', '֞', 'ო', 'ш', 'ҏ', 'գ', 'ղ', 'ե', 'գ', 'է', 'ը',
'թ', 'ժ', 'ի', 'լ', 'խ', 'ծ', 'կ', 'h', 'ձ', 'ղ', 'ճ', 'մ', 'յ',
'ն', '2', 'ո', 'չ', 'պ', 'ջ', 'ռ', 'u', 'վ', 'ш', 'ր', 'g', 'ւ',
'փ', 'ք', 'o', 'ֆ', 'և', 'ֈ', ':', '-', '\u058b', '\u058c', '֍',
'֎', '֏', '\u0590', '֑', '֒', '֓', '֔', '֕', '֖', '֗', '֘', '֙', '֚', '֛',
'֜', '֝', '֞', '֟', '֠', '֡', '֢', '֣', '֤', '֥', '֦', '֧', '֨', '֩', '֪', '֫',
'֬', '֭', '֮', '֯', 'ְ', 'ֱ', 'ֲ', 'ֳ', 'ִ', 'ֵ', 'ֶ', 'ַ', 'ָ', 'ֹ', 'ֺ', 'ֻ',
'ּ', 'ֽ', '־', 'ֿ', '׀', 'ׁ', 'ׂ', '׃', 'ׄ', 'ׅ', '׆', 'ׇ', '\u05c8', '\u05c9', '\u
5ca', '\u05cb', '\u05cc', '\u05cd', '\u05ce', '\u05cf', 'א','ב' ,'ג'
,'ד' ,'ה' ,'ו' ,'ז' ,'ח' ,'ט' ,'י' ,'ך' ,'כ' ,'ל' ,'ם' ,'מ'
ת' ,'ש' ,'ר' ,'ק' ,'צ' ,'ץ' ,'פ' ,'ף' ,'ע' ,'ס' ,'נ' ,'ן', '\u05e
b', '\u05ec', '\u05ed', '\u05ee', '״' ,'׳' ,'ׯ' ,'װ' ,'ױ ,'', '\u
05f5', '\u05f6', '\u05f7', '\u05f8', '\u05f9', '\u05fa', '\u05fb',
'\u05fc', '\u05fd', '\u05fe', '\u05ff', '\u0600', '\u0601', '\u060
', '\u0603', '\u0604', '\u0605', '؆', '؇', ' ','ف' ,'‰' ,'٪' ,'ق
ٔ' ,'ٕ', 'ٖ', 'ٗ', 'ٚ', 'ٛ', 'ٜ', 'ٝ']
```

In [7]:
```
1 print([chr(v) for v in range(1556,2048)])
```

```
,'؟' ,'؞' ,'u061c', '\u061d\' ,'ؠ' ,'ء' ,'آ' ,'أ' ,'ؤ' ,'إ' ,'ئ' ,'ا' ,'ب' ,'ة' ,'ت' ,'ث' ,'ج'
,'ح' ,'خ' ,'د' ,'ذ' ,'ر' ,'ز' ,'س' ,'ش' ,'ص' ,'ض' ,'ط' ,'ظ' ,'ع' ,'غ' ,'ـ' ,'ف' ,'ق' ,'ك' ,'ل' ,'م' ,'ن'
,'ه' ,'و' ,'ى' ,'ي' ,'ً' ,'ٌ' ,'ٍ' ,'َ' ,'ُ' ,'ِ' ,'ّ' ,'ْ' ,'ٓ' ,'ٔ' ,'ٕ' ,'٠' ,'١' ,'٢' ,'٣' ,'٤' ,'٥'
,'٦' ,'٧' ,'٨' ,'٩' ,'٪' ,'٫' ,'٬' ,'٭' ,'ٮ' ,'ٯ' ,'ٰ' ,'ٱ' ,'ٲ' ,'ٳ' ,'ٴ' ,'ٵ' ,'ٶ' ,'ٷ' ,'ٸ' ,'ٹ'
,'ٺ' ,'ٻ' ,'ټ' ,'ٽ' ,'پ' ,'ٿ' ,'ڀ' ,'ځ' ,'ڂ' ,'ڃ' ,'ڄ' ,'څ' ,'چ' ,'ڇ' ,'ڈ' ,'ډ' ,'ڊ' ,'ڋ' ,'ڌ' ,'ڍ'
,'ڎ' ,'ڏ' ,'ڐ' ,'ڑ' ,'ڒ' ,'ړ' ,'ڔ' ,'ڕ' ,'ږ' ,'ڗ' ,'ژ' ,'ڙ' ,'ښ' ,'ڛ' ,'ڜ' ,'ڝ' ,'ڞ' ,'ڟ' ,'ڠ' ,'ڡ'
,'ڢ' ,'ڣ' ,'ڤ' ,'ڥ' ,'ڦ' ,'ڧ' ,'ڨ' ,'ک' ,'ڪ' ,'ګ' ,'ڬ' ,'ڭ' ,'ڮ' ,'گ' ,'ڰ' ,'ڱ' ,'ڲ' ,'ڳ' ,'ڴ' ,'ڵ'
,'ڶ' ,'ڷ' ,'ڸ' ,'ڹ' ,'ں' ,'ڻ' ,'ڼ' ,'ڽ' ,'ھ' ,'ڿ' ,'ۀ' ,'ہ' ,'ۂ' ,'ۃ' ,'ۄ' ,'ۅ' ,'ۆ' ,'ۇ' ,'ۈ' ,'ۉ'
,'ۊ' ,'ۋ' ,'ی' ,'ۍ' ,'ێ' ,'ۏ' ,'ې' ,'ۑ' ,'ے' ,'ۓ' ,'ە' ,'u06dd\' ,'ۮ' ,'ۯ' ,'ۺ' ,'ۻ' ,'ۼ' ,'۽' ,'۾'
,'ۿ' ,'ܐ' ,'ܑ' ,'ܒ' ,'ܓ' ,'ܔ' ,'ܕ' ,'ܖ' ,'ܗ' ,'ܘ' ,'ܙ' ,'ܚ' ,'ܛ' ,'ܜ' ,'ܝ' ,'ܞ' ,'ܟ' ,'ܠ' ,'ܡ' ,'ܢ'
,'ܣ' ,'ܤ' ,'ܥ' ,'ܦ' ,'ܧ' ,'ܨ' ,'ܩ' ,'ܪ' ,'ܫ' ,'ܬ' ,'ܭ' ,'ܮ' ,'ܯ' ,'ܰ' ,'ܱ' ,'ܲ' ,'ܳ' ,'ܴ' ,'ܵ' ,'ܶ'
,'ܷ' ,'ܸ' ,'ܹ' ,'ܺ' ,'ܻ' ,'ܼ' ,'ܽ' ,'ܾ' ,'ܿ' ,'݀' ,'݁' ,'݂' ,'݃' ,'݄' ,'݅' ,'݆' ,'݇' ,'݈' ,'݉' ,'ݍ'
,'ݎ' ,'ݏ' ,'ݐ' ,'ݑ' ,'ݒ' ,'ݓ' ,'ݔ' ,'ݕ' ,'ݖ' ,'ݗ' ,'ݘ' ,'ݙ' ,'ݚ' ,'ݛ' ,'ݜ' ,'ݝ' ,'ݞ' ,'ݟ' ,'ݠ' ,'ݡ'
,'ݢ' ,'ݣ' ,'ݤ' ,'ݥ' ,'ݦ' ,'ݧ' ,'ݨ' ,'ݩ' ,'ݪ' ,'ݫ' ,'ݬ' ,'ݭ' ,'ݮ' ,'ݯ' ,'ݰ' ,'ݱ' ,'ݲ' ,'ݳ' ,'ݴ' ,'ݵ'
,'ݶ' ,'ݷ' ,'ݸ' ,'ݹ' ,'ݺ' ,'ݻ' ,'ݼ' ,'ݽ' ,'ݾ' ,'ݿ' ,'ހ' ,'ށ' ,'ނ' ,'ރ' ,'ބ' ,'ޅ' ,'ކ' ,'އ' ,'ވ' ,' މ'
,'ފ' ,'ދ' ,'ތ' ,'ލ' ,'ގ' ,'ޏ' ,'ސ' ,'ޑ' ,'ޒ' ,'ޓ' ,'ޔ' ,'ޕ' ,'ޖ' ,'ޗ' ,'ޘ' ,'ޙ' ,'ޚ' ,'ޛ' ,'ޜ' ,'ޝ'
,'ޞ' ,'ޟ' ,'ޠ' ,'ޡ' ,'ޢ' ,'ޣ' ,'ޤ' ,'ޥ' ,'ަ' ,'ާ' ,'ި' ,'ީ' ,'ު' ,'ޫ' ,'ެ' ,'ޭ' ,'ޮ' ,'ޯ' ,'ް' ,'ޱ'
,'۰' ,'۱' ,'۲' ,'۳' ,'۴' ,'۵' ,'۶' ,'۷' ,'۸' ,'۹' ,'ۺ' ,'ۻ' ,'ۼ' ,'۽' ,'۾' ,'ۿ' ,'ऀ' ,'ँ' ,'ं' ,'ः'
,'ऄ' ,'अ' ,'आ' ,'इ' ,'ई' ,'उ' ,'ऊ' ,'ऋ' ,'ऌ' ,'u070e', '\u070f\' ,'ऐ' ,'ऑ' ,'ऒ' ,'ओ' ,'औ' ,'क' ,'ख'
,'ग' ,'घ' ,'ङ' ,'च' ,'छ' ,'ज' ,'झ' ,'ञ' ,'ट' ,'ठ' ,'ड' ,'ढ' ,'ण' ,'त' ,'थ' ,'द' ,'ध' ,'न' ,'ऩ' ,'प'
,'फ' ,'ब' ,'भ' ,'म' ,'य' ,'र' ,'ऱ' ,'ल' ,'ळ' ,'ऴ' ,'व' ,'श' ,'ष' ,'स' ,'ह' ,'ऺ' ,'ऻ' ,'़' ,'ऽ' ,'ा'
,'ि' ,'ी' ,'ु' ,'ू' ,'ृ' ,'ॄ' ,'ॅ' ,'ॆ' ,'े' ,'ै' ,'ॉ' ,'ॊ' ,'ो' ,'ौ' ,'्' ,'ॎ' ,'ॏ' ,'ॐ' ,'॑' ,'॒'
,'॓' ,'॔' ,'ॕ' ,'ॖ' ,'ॗ' ,'क़' ,'ख़' ,'ग़' ,'ज़' ,'ड़' ,'ढ़' ,'फ़' ,'य़' ,'ॠ' ,'ॡ' ,'ॢ' ,'ॣ' ,'।' ,'॥' ,'०'
,'१' ,'२' ,'३' ,'४' ,'५' ,'६' ,'७' ,'८' ,'९' ,'॰' ,'ॱ' ,'ॲ' ,'ॳ' ,'ॴ' ,'ॵ' ,'ॶ' ,'ॷ' ,'ॸ' ,'ॹ' ,'ॺ'
,'ॻ' ,'ॼ' ,'ॽ' ,'ॾ' ,'ॿ' ,'ঀ' ,'ঁ' ,'ং' ,'ঃ' ,'অ' ,'আ' ,'ই' ,'ঈ' ,'উ' ,'ঊ' ,'ঋ' ,'ঌ' ,'঍' ,'এ' ,'ঐ'
,'঑' ,'ও' ,'ঔ' ,'ক' ,'খ' ,'গ' ,'ঘ' ,'ঙ' ,'চ' ,'ছ' ,'জ' ,'ঝ' ,'ঞ' ,'ট' ,'ঠ' ,'ড' ,'ঢ' ,'ণ' ,'ত' ,'থ'
,'দ' ,'ধ' ,'ন' ,'঩' ,'প' ,'ফ' ,'ব' ,'ভ' ,'ম' ,'য' ,'র' ,'঱' ,'ল' ,'঳' ,'঴' ,'঵' ,'শ' ,'ষ' ,'স' ,'হ'
,'঺' ,'঻' ,'়' ,'ঽ' ,'া' ,'ি' ,'ী' ,'ু' ,'ূ' ,'ৃ' ,'ৄ' ,'৅' ,'৆' ,'ে' ,'ৈ' ,'৉' ,'৊' ,'ো' ,'ৌ' ,'্'
,'ৎ' ,'৏' ,'৐' ,'৑' ,'৒' ,'৓' ,'৔' ,'৕' ,'৖' ,'ৗ' ,'৘' ,'৙' ,'৚' ,'৛' ,'ড়' ,'ঢ়' ,'য়' ,'ৠ' ,'ৡ' ,'ৢ'
,'ৣ' ,'৤' ,'৥' ,'০' ,'১' ,'২' ,'৩' ,'৪' ,'৫' ,'৬' ,'৭' ,'৮' ,'৯' ,'ৰ' ,'ৱ' ,'৲' ,'৳' ,'৴' ,'৵' ,'৶'
,'৷' ,'৸' ,'৹' ,'৺' ,'৻' ,'ৼ' ,'৽' ,'৾' ,'৿' ,'਀' ,'ਁ' ,'ਂ' ,'ਃ' ,'਄' ,'ਅ' ,'ਆ' ,'ਇ' ,'ਈ' ,'ਉ' ,'ਊ'
,'਋' ,'਌' ,'਍' ,'ਏ' ,'ਐ' ,'਑' ,'਒' ,'ਓ' ,'ਔ' ,'ਕ' ,'ਖ' ,'ਗ' ,'ਘ' ,'ਙ' ,'ਚ' ,'ਛ' ,'ਜ' ,'ਝ' ,'ਞ' ,'ਟ'
,'ਠ' ,'ਡ' ,'ਢ' ,'ਣ' ,'ਤ' ,'ਥ' ,'ਦ' ,'ਧ' ,'ਨ' ,'਩' ,'ਪ' ,'ਫ' ,'ਬ' ,'ਭ' ,'ਮ' ,'ਯ' ,'ਰ' ,'਱' ,'ਲ' ,'ਲ਼'
,'਴' ,'ਵ' ,'ਸ਼' ,'਷' ,'ਸ' ,'ਹ' ,'਺' ,'਻' ,'਼' ,'਽' ,'ਾ' ,'ਿ' ,'ੀ' ,'ੁ' ,'ੂ' ,'੃' ,'੄' ,'੅' ,'੆' ,'ੇ'
,'ੈ' ,'੉' ,'੊' ,'ੋ' ,'ੌ' ,'੍' ,'ੑ' ,'੒' ,'੓' ,'੔' ,'ੜ' ,'੝' ,'ਫ਼' ,'੟' ,'੠' ,'੡' ,'੢' ,'੣' ,'੤' ,'੥'
,'੦' ,'੧' ,'੨' ,'੩' ,'੪' ,'੫' ,'੬' ,'੭' ,'੮' ,'੯' ,'ੰ' ,'ੱ' ,'ੲ' ,'ੳ' ,'ੴ' ,'ੵ' ,'੶' ,'੷' ,'੸' ,'੹'
,'੺' ,'੻' ,'੼' ,'੽' ,'੾' ,'੿' ,'઀' ,'ઁ' ,'ં' ,'ઃ' ,'઄' ,'અ' ,'આ' ,'ઇ' ,'ઈ' ,'ઉ' ,'ઊ' ,'ઋ' ,'ઌ' ,'ઍ'
,'઎' ,'એ' ,'ઐ' ,'ઑ' ,'઒' ,'ઓ' ,'ઔ' ,'ક' ,'ખ' ,'ગ' ,'ઘ' ,'ઙ' ,'ચ' ,'છ' ,'જ' ,'ઝ' ,'ઞ' ,'ટ' ,'ઠ' ,'ડ'
,'ઢ' ,'ણ' ,'ત' ,'થ' ,'દ' ,'ધ' ,'ન' ,'઩' ,'પ' ,'ફ' ,'બ' ,'ભ' ,'મ' ,'ય' ,'ર' ,'઱' ,'લ' ,'ળ' ,'઴' ,'વ'
,'શ' ,'ષ' ,'સ' ,'હ' ,'઺' ,'઻' ,'઼' ,'ઽ' ,'ા' ,'િ' ,'ી' ,'ુ' ,'ૂ' ,'ૃ' ,'ૄ' ,'ૅ' ,'ે' ,'ૈ' ,'ૉ' ,'ો'
,'ૌ' ,'્' ,'૎' ,'ૐ' ,'૑' ,'૒' ,'૓' ,'૔' ,'૕' ,'૖' ,'૗' ,'૘' ,'૙' ,'૚' ,'૛' ,'૜' ,'૝' ,'૞' ,'૟' ,'ૠ'
,'ૡ' ,'ૢ' ,'ૣ' ,'૤' ,'૥' ,'૦' ,'૧' ,'૨' ,'૩' ,'૪' ,'૫' ,'૬' ,'૭' ,'૮' ,'૯' ,'૰' ,'૱' ,'ૻ' ,'ૼ' ,'૽'
,'૾' ,'૿' ,'଀' ,'ଁ' ,'ଂ' ,'ଃ' ,'଄' ,'ଅ' ,'ଆ' ,'ଇ' ,'ଈ' ,'ଉ' ,'ଊ' ,'ଋ' ,'ଌ' ,'଍' ,'ଏ' ,'ଐ' ,'଑' ,'ଓ'
,'ଔ' ,'କ' ,'ଖ' ,'ଗ' ,'ଘ' ,'ଙ' ,'ଚ' ,'ଛ' ,'ଜ' ,'ଝ' ,'ଞ' ,'ଟ' ,'ଠ' ,'ଡ' ,'ଢ' ,'ଣ' ,'ତ' ,'ଥ' ,'ଦ' ,'ଧ'
,'ନ' ,'଩' ,'ପ' ,'ଫ' ,'ବ' ,'ଭ' ,'ମ' ,'ଯ' ,'ର' ,'଱' ,'ଲ' ,'ଳ' ,'଴' ,'ଵ' ,'ଶ' ,'ଷ' ,'ସ' ,'ହ' ,'଺' ,'଻'
,'଼' ,'ଽ' ,'ା' ,'ି' ,'ୀ' ,'ୁ' ,'ୂ' ,'ୃ' ,'ୄ' ,'୅' ,'୆' ,'େ' ,'ୈ' ,'୉' ,'୊' ,'ୋ' ,'ୌ' ,'୍' ,'୎' ,'୏'
,'୐' ,'୑' ,'୒' ,'୓' ,'୔' ,'୕' ,'ୖ' ,'ୗ' ,'୘' ,'୙' ,'୚' ,'୛' ,'ଡ଼' ,'ଢ଼' ,'ୟ' ,'ୠ' ,'ୡ' ,'ୢ' ,'ୣ' ,'୤'
,'୥' ,'୦' ,'୧' ,'୨' ,'୩' ,'୪' ,'୫' ,'୬' ,'୭' ,'୮' ,'୯' ,'୰' ,'ୱ' ,'୲' ,'୳' ,'୴' ,'୵' ,'୶' ,'୷' ,'஀'
,'ஂ' ,'ஃ' ,'அ' ,'ஆ' ,'இ' ,'ஈ' ,'உ' ,'ஊ' ,'எ' ,'ஏ' ,'ஐ' ,'ஒ' ,'ஓ' ,'ஔ' ,'க' ,'ங' ,'ச' ,'ஜ' ,'ஞ' ,'ட'
,'ண' ,'த' ,'ந' ,'ன' ,'ப' ,'ம' ,'ய' ,'ர' ,'ற' ,'ல' ,'ள' ,'ழ' ,'வ' ,'ஶ' ,'ஷ' ,'ஸ' ,'ஹ' ,'஺' ,'஻'
,'஼' ,'஽' ,'ா' ,'ி' ,'ீ' ,'ு' ,'ூ' ,'ெ' ,'ே' ,'ை' ,'ொ' ,'ோ' ,'ௌ' ,'்' ,'ௐ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ'
,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ' ,'ௗ'
u07b2', '\u07b3', '\u07b4', '\u07b\' ,'ఄ' ,'అ' ,'ఆ' ,'ఇ' ,'ఈ' ,'ఉ' ,'ఊ' ,'ఋ' ,'ఌ' ,'఍' ,'ఎ' ,'ఏ' ,'ఐ' ,'఑' ,'ఒ' ,'ఓ'
5', '\u07b6', '\u07b7', '\u07b8', '\u07b9', '\u07ba', '\u07bb', '\u
,'ఴ' ,'వ' ,'శ' ,'ష' ,'స' ,'హ' ,'఺' ,'఻' ,'఼' ,'ఽ' ,'ా' ,'ి' ,'ీ' ,'ు' ,'ూ' ,'ృ' ,'ౄ' ,'౅' ,'ె' ,'ే' ,'ై'
,'ౘ' ,'ౙ' ,'ౚ' ,'౛' ,'ౝ' ,'ో' ,'ౌ' ,'్' ,'౎' ,'౏' ,'ౕ' ,'ౖ' ,'౗' ,'ౘ' ,'ౙ' ,'ౚ' ,'ౠ' ,'ౡ' ,'ౢ' ,'ౣ'
,'౤' ,'౥' ,'౦' ,'౧' ,'౨' ,'౩' ,'౪' ,'౫' ,'౬' ,'౭' ,'౮' ,'౯' ,'౰' ,'౱' ,'౲' ,'౳' ,'౴' ,'౵' ,'౶' ,'౷'
,'౸' ,'౹' ,'౺' ,'౻' ,'౼' ,'౽' ,'౾' ,'౿' ,'ಀ' ,'ಁ' ,'ಂ' ,'ಃ' ,'಄' ,'ಅ' ,'ಆ' ,'ಇ' ,'ಈ' ,'ಉ' ,'ಊ' ,'ಋ'
,'ಌ' ,'಍' ,'ಎ' ,'ಏ' ,'ಐ' ,'಑' ,'ಒ' ,'ಓ' ,'ಔ' ,'ಕ' ,'ಖ' ,'ಗ' ,'ಘ' ,'ಙ' ,'ಚ' ,'ಛ' ,'ಜ' ,'ಝ' ,'ಞ' ,'ಟ'
,'ಠ' ,'ಡ' ,'ಢ' ,'ಣ' ,'ತ' ,'ಥ' ,'ದ' ,'ಧ' ,'ನ' ,'಩' ,'ಪ' ,'ಫ' ,'ಬ' ,'ಭ' ,'ಮ' ,'ಯ' ,'ರ' ,'ಱ' ,'ಲ' ,'ಳ'
,'಴' ,'ವ' ,'ಶ' ,'ಷ' ,'ಸ' ,'ಹ' ,'಺' ,'಻' ,'಼' ,'ಽ' ,'ಾ' ,'ಿ' ,'ೀ' ,'ು' ,'ೂ' ,'ೃ' ,'ೄ' ,'೅' ,'ೆ' ,'ೇ'
,'ೈ' ,'೉' ,'ೊ' ,'ೋ' ,'ೌ' ,'್' ,'೎' ,'೏' ,'೐' ,'೑' ,'೒' ,'೓' ,'೔' ,'ೕ' ,'ೖ' ,'೗' ,'೘' ,'೙' ,'೚' ,'೛'
,'೜' ,'ೝ' ,'ೞ' ,'೟' ,'ೠ' ,'ೡ' ,'ೢ' ,'ೣ' ,'೤' ,'೥' ,'೦' ,'೧' ,'೨' ,'೩' ,'೪' ,'೫' ,'೬' ,'೭' ,'೮' ,'೯'
,'೰' ,'ೱ' ,'ೲ' ,'ೳ' ,'೴' ,'೵' ,'೶' ,'೷' ,'೸' ,'೹' ,'೺' ,'೻' ,'೼' ,'೽' ,'೾' ,'೿' ,'ഀ' ,'ഁ' ,'ം' ,'ഃ'
,'ഄ' ,'അ' ,'ആ' ,'ഇ' ,'ഈ' ,'ഉ' ,'ഊ' ,'ഋ' ,'ഌ' ,'഍' ,'എ' ,'ഏ' ,'ഐ' ,'഑' ,'ഒ' ,'ഓ' ,'ഔ' ,'ക' ,'ഖ'
,'ഗ' ,'ഘ' ,'ങ' ,'ച' ,'ഛ' ,'ജ' ,'ഝ' ,'ഞ' ,'ട' ,'ഠ' ,'ഡ' ,'ഢ' ,'ണ' ,'ത' ,'ഥ' ,'ദ' ,'ധ' ,'ന' ,'ഩ' ,'പ'
,'ഫ' ,'ബ' ,'ഭ' ,'മ' ,'യ' ,'ര' ,'റ' ,'ല' ,'ള' ,'ഴ' ,'വ' ,'ശ' ,'ഷ' ,'സ' ,'ഹ' ,'ഺ' ,'഻' ,'഼' ,'ഽ' ,'ാ'
,'ി' ,'ീ' ,'ു' ,'ൂ' ,'ൃ' ,'ൄ' ,'൅' ,'െ' ,'േ' ,'ൈ' ,'൉' ,'ൊ' ,'ോ' ,'ൌ' ,'്' ,'ൎ' ,'൏' ,'ൔ' ,'ൕ' ,'ൖ'
,'ൗ' ,'൘' ,'൙' ,'൚' ,'൛' ,'൜' ,'൝' ,'൞' ,'ൟ' ,'ൠ' ,'ൡ' ,'ൢ' ,'ൣ' ,'൤' ,'൥' ,'൦' ,'൧' ,'൨' ,'൩' ,'൪'
,'൫' ,'൬' ,'൭' ,'൮' ,'൯' ,'൰' ,'൱' ,'൲' ,'൳' ,'൴' ,'൵' ,'൶' ,'൷' ,'൸' ,'൹' ,'ൺ' ,'ൻ' ,'ർ' ,'ൽ' ,'ൾ'
,'ൿ' ,'඀' ,'ඁ' ,'ං' ,'ඃ' ,'඄' ,'අ' ,'ආ' ,'ඇ' ,'ඈ' ,'ඉ' ,'ඊ' ,'උ' ,'ඌ' ,'ඍ' ,'ඎ' ,'ඏ' ,'ඐ' ,'එ' ,'ඒ'
,'ඓ' ,'ඔ' ,'ඕ' ,'ඖ' ,'඗' ,'඘' ,'඙' ,'ක' ,'ඛ' ,'ග' ,'ඝ' ,'ඞ' ,'ඟ' ,'ච' ,'ඡ' ,'ජ' ,'ඣ' ,'ඤ' ,'ඥ' ,'ඦ'
,'ට' ,'ඨ' ,'ඩ' ,'ඪ' ,'ණ' ,'ඬ' ,'ත' ,'ථ' ,'ද' ,'ධ' ,'න' ,'ඳ' ,'ප' ,'ඵ' ,'බ' ,'භ' ,'ම' ,'ඹ' ,'ය' ,'ර'
,'඼' ,'ල' ,'ළ' ,'ෆ' ,'඾' ,'඿' ,'ව' ,'ශ' ,'ෂ' ,'ස' ,'හ' ,'ළ' ,'ෆ' ,'඾' ,'඿' ,'ව' ,'ශ' ,'ෂ' ,'ස' ,'හ'
,'ෑ' ,'ි' ,'ී' ,'ු' ,'ූ' ,'ෘ' ,'ෙ' ,'ේ' ,'ෛ' ,'ො' ,'ෝ' ,'ෞ' ,'ෟ' ,'්' ,'෡' ,'෢' ,'෣' ,'෤' ,'෥' ,'෦'
['ษ' ,'ฺ' ,'๏' ,'u07fb', '\u07fc\' ,'฿'
```

Etcètera. Per a valors superiors, trobem símbols d'altres alfabets (per exemple, orientals).

In [8]:
```
1 print([chr(v) for v in range(23456,23600)])
```

```
['宠', '审', '客', '宣', '室', '宥', '宦', '宧', '寇', '寀', '宪',
'宫', '宬', '宻', '宮', '孳', '宰', '宲', '寀', '害', '宴', '宵', '家',
'寀', '宸', '容', '寃', '密', '寇', '宽', '宾', '宿', '寀', '隶', '寂',
'寃', '寄', '寅', '密', '寇', '青', '雀', '貞', '寨', '富', '盛', '寎',
'窦', '寐', '寝', '寒', '寓', '寔', '寧', '寖', '窩', '真', '寙', '害',
'寬', '寧', '寝', '寞', '察', '寠', '寡', '寢', '寤', '寤', '寥', '實',
'寧', '寨', '審', '寫', '寫', '寬', '惠', '寮', '寓', '寰', '寤', '寖',
'寶', '親', '寵', '寶', '豐', '寸', '对', '寺', '寻', '导', '孚', '对',
'寿', '尌', '封', '專', '尃', '射', '尅', '将', '將', '專', '尉', '尊',
'尋', '尉', '對', '導', '小', '尐', '少', '尒', '尔', '尔', '尒', '尖',
'尗', '尘', '尙', '尚', '尒', '尒', '尝', '尞', '尟', '尠', '尰', '尢',
'尣', '尤', '尥', '尦', '尧', '尨', '尪', '尬', '尷', '尬', '尭', '尮',
'尷']
```

I per a valors més grans, hi ha caràcters interpretats en format unicode de 8 caràcters hexadecimals (per exemple, \U0001000c).

In [9]:
```
1 print([chr(v) for v in range(2^16,2^16+256)])
```

```
['𐽔', '𐽕', '𐽖', '𐽗', '𐽘', '𐽙', '𐽚', '𐽛', '𐽜', '𐽝', '𐽞', '𐽟', '\U00
01000c', '𐽠', '𐽡', '𐽢', '𐽣', '𐽤', '𐽥', '𐽦', '𐽧', '𐽨', '𐽩', '𐽪', '𐽫',
'𐽬', '𐽭', '𐽮', '𐽯', '𐽰', '𐽱', '𐽲', '𐽳', '𐽴', '𐽵', '𐽶', '𐽷', '𐽸',
'𐽹', '𐽺', '\U00010027', '𐽼', '𐽽', '𐽾', '𐽿', '𐾀', '𐾁', '𐾂', '𐾃',
'𐾄', '𐾅', '𐾆', '𐾇', '𐾈', '𐾉', '𐾊', '𐾋', '𐾌', '𐾍', '\U0001003
b', '𐾎', '𐾏', '\U0001003e', '𐾐', '𐾑', '𐾒', '𐾓', '𐾔', '𐾕', '𐾖',
'𐾗', '𐾘', '𐾙', '𐾚', '𐾛', '𐾜', '𐾝', '𐾞', '\U0001004e', '\U0001004
f', '𐾟', '𐾠', '𐾡', '𐾢', '𐾣', '𐾤', '𐾥', '𐾦', '𐾧', '𐾨', '𐾩', '𐾪',
'𐾫', '𐾬', '\U0001005e', '\U0001005f', '\U00010060', '\U00010061',
'\U00010062', '\U00010063', '\U00010064', '\U00010065', '\U0001006
6', '\U00010067', '\U00010068', '\U00010069', '\U0001006a', '\U0001
006b', '\U0001006c', '\U0001006d', '\U0001006e', '\U0001006f', '\U0
0010070', '\U00010071', '\U00010072', '\U00010073', '\U00010074',
'\U00010075', '\U00010076', '\U00010077', '\U00010078', '\U0001007
9', '\U0001007a', '\U0001007b', '\U0001007c', '\U0001007d', '\U0001
007e', '\U0001007f', '𐾀', '𐾁', '𐾂', '𐾃', '𐾄', '𐾅', '𐾆', '𐾇',
'𐾈', '𐾉', '𐾊', '𐾋', '𐾌', '𐾍', '𐾎', '𐾏', '𐾐', '𐾑', '𐾒', '𐾓',
'𐾔', '𐾕', '𐾖', '𐾗', '𐾘', '𐾙', '𐾚', '𐾛', '𐾜', '𐾝', '𐾞', '𐾟',
'𐾠', '𐾡', '𐾢', '𐾣', '𐾤', '𐾥', '𐾦', '𐾧', '𐾨', '𐾩', '𐾪', '𐾫',
'𐾬', '𐾭', '𐾮', '𐾯', '𐾰', '𐾱', '𐾲', '𐾳', '𐾴', '𐾵', '𐾶', '𐾷',
'𐾸', '𐾹', '𐾺', '𐾻', '𐾼', '𐾽', '𐾾', '𐾿', '𐿀', '𐿁', '𐿂', '𐿃',
'𐿄', '𐿅', '𐿆', '𐿇', '𐿈', '𐿉', '𐿊', '𐿋', '𐿌', '𐿍', '𐿎', '𐿏',
'𐿐', '𐿑', '𐿒', '𐿓', '𐿔', '𐿕', '𐿖', '𐿗', '𐿘', '𐿙', '𐿚', '𐿛',
'𐿜', '𐿝', '𐿞', '𐿟', '𐿠', '𐿡', '𐿢', '𐿣', '𐿤', '𐿥', '𐿦', '𐿧',
'𐿨', '𐿩', '𐿪', '𐿫', '𐿬', '𐿭', '𐿮', '𐿯', '𐿰', '𐿱', '𐿲', '𐿳',
'𐿴', '𐿵', '𐿶', '𐿷', '𐿸', '𐿹', '𐿺', '𐿻', '𐿼', '𐿽', '\U000100fb',
'\U000100fc', '\U000100fd', '\U000100fe', '\U000100ff']
```

Mirem-nos els darrers possibles (el darrer és \U0010ffff):

In [10]:
```
1 print([chr(v) for v in range(16^5+16^4-256,16^5+16^4)])
```

```
['\U0010ff00', '\U0010ff01', '\U0010ff02', '\U0010ff03', '\U0010ff0
4', '\U0010ff05', '\U0010ff06', '\U0010ff07', '\U0010ff08', '\U0010
ff09', '\U0010ff0a', '\U0010ff0b', '\U0010ff0c', '\U0010ff0d', '\U0
010ff0e', '\U0010ff0f', '\U0010ff10', '\U0010ff11', '\U0010ff12',
'\U0010ff13', '\U0010ff14', '\U0010ff15', '\U0010ff16', '\U0010ff1
7', '\U0010ff18', '\U0010ff19', '\U0010ff1a', '\U0010ff1b', '\U0010
ff1c', '\U0010ff1d', '\U0010ff1e', '\U0010ff1f', '\U0010ff20', '\U0
010ff21', '\U0010ff22', '\U0010ff23', '\U0010ff24', '\U0010ff25',
'\U0010ff26', '\U0010ff27', '\U0010ff28', '\U0010ff29', '\U0010ff2
a', '\U0010ff2b', '\U0010ff2c', '\U0010ff2d', '\U0010ff2e', '\U0010
ff2f', '\U0010ff30', '\U0010ff31', '\U0010ff32', '\U0010ff33', '\U0
010ff34', '\U0010ff35', '\U0010ff36', '\U0010ff37', '\U0010ff38',
'\U0010ff39', '\U0010ff3a', '\U0010ff3b', '\U0010ff3c', '\U0010ff3
d', '\U0010ff3e', '\U0010ff3f', '\U0010ff40', '\U0010ff41', '\U0010
ff42', '\U0010ff43', '\U0010ff44', '\U0010ff45', '\U0010ff46', '\U0
010ff47', '\U0010ff48', '\U0010ff49', '\U0010ff4a', '\U0010ff4b',
'\U0010ff4c', '\U0010ff4d', '\U0010ff4e', '\U0010ff4f', '\U0010ff5
0', '\U0010ff51', '\U0010ff52', '\U0010ff53', '\U0010ff54', '\U0010
ff55', '\U0010ff56', '\U0010ff57', '\U0010ff58', '\U0010ff59', '\U0
010ff5a', '\U0010ff5b', '\U0010ff5c', '\U0010ff5d', '\U0010ff5e',
'\U0010ff5f', '\U0010ff60', '\U0010ff61', '\U0010ff62', '\U0010ff6
3', '\U0010ff64', '\U0010ff65', '\U0010ff66', '\U0010ff67', '\U0010
ff68', '\U0010ff69', '\U0010ff6a', '\U0010ff6b', '\U0010ff6c', '\U0
010ff6d', '\U0010ff6e', '\U0010ff6f', '\U0010ff70', '\U0010ff71',
'\U0010ff72', '\U0010ff73', '\U0010ff74', '\U0010ff75', '\U0010ff7
6', '\U0010ff77', '\U0010ff78', '\U0010ff79', '\U0010ff7a', '\U0010
ff7b', '\U0010ff7c', '\U0010ff7d', '\U0010ff7e', '\U0010ff7f', '\U0
010ff80', '\U0010ff81', '\U0010ff82', '\U0010ff83', '\U0010ff84',
'\U0010ff85', '\U0010ff86', '\U0010ff87', '\U0010ff88', '\U0010ff8
9', '\U0010ff8a', '\U0010ff8b', '\U0010ff8c', '\U0010ff8d', '\U0010
ff8e', '\U0010ff8f', '\U0010ff90', '\U0010ff91', '\U0010ff92', '\U0
010ff93', '\U0010ff94', '\U0010ff95', '\U0010ff96', '\U0010ff97',
'\U0010ff98', '\U0010ff99', '\U0010ff9a', '\U0010ff9b', '\U0010ff9
c', '\U0010ff9d', '\U0010ff9e', '\U0010ff9f', '\U0010ffa0', '\U0010
ffa1', '\U0010ffa2', '\U0010ffa3', '\U0010ffa4', '\U0010ffa5', '\U0
010ffa6', '\U0010ffa7', '\U0010ffa8', '\U0010ffa9', '\U0010ffaa',
'\U0010ffab', '\U0010ffac', '\U0010ffad', '\U0010ffae', '\U0010ffa
f', '\U0010ffb0', '\U0010ffb1', '\U0010ffb2', '\U0010ffb3', '\U0010
ffb4', '\U0010ffb5', '\U0010ffb6', '\U0010ffb7', '\U0010ffb8', '\U0
010ffb9', '\U0010ffba', '\U0010ffbb', '\U0010ffbc', '\U0010ffbd',
'\U0010ffbe', '\U0010ffbf', '\U0010ffc0', '\U0010ffc1', '\U0010ffc
2', '\U0010ffc3', '\U0010ffc4', '\U0010ffc5', '\U0010ffc6', '\U0010
ffc7', '\U0010ffc8', '\U0010ffc9', '\U0010ffca', '\U0010ffcb', '\U0
010ffcc', '\U0010ffcd', '\U0010ffce', '\U0010ffcf', '\U0010ffd0',
'\U0010ffd1', '\U0010ffd2', '\U0010ffd3', '\U0010ffd4', '\U0010ffd
5', '\U0010ffd6', '\U0010ffd7', '\U0010ffd8', '\U0010ffd9', '\U0010
ffda', '\U0010ffdb', '\U0010ffdc', '\U0010ffdd', '\U0010ffde', '\U0
010ffdf', '\U0010ffe0', '\U0010ffe1', '\U0010ffe2', '\U0010ffe3',
'\U0010ffe4', '\U0010ffe5', '\U0010ffe6', '\U0010ffe7', '\U0010ffe
8', '\U0010ffe9', '\U0010ffea', '\U0010ffeb', '\U0010ffec', '\U0010
ffed', '\U0010ffee', '\U0010ffef', '\U0010fff0', '\U0010fff1', '\U0
010fff2', '\U0010fff3', '\U0010fff4', '\U0010fff5', '\U0010fff6',
'\U0010fff7', '\U0010fff8', '\U0010fff9', '\U0010fffa', '\U0010fff
b', '\U0010fffc', '\U0010fffd', '\U0010fffe', '\U0010ffff']
```

Però si intentem veure'n un de posterior, obtenim un error (òbviament).

In [11]:
```
1 print(chr(16^5+16^4))
```

```
---------------------------------------------------------------------
--------
ValueError                                Traceback (most recent ca
ll last)
/tmp/ipykernel_29741/2710118598.py in <module>
----> 1 print(chr(Integer(16)**Integer(5)+Integer(16)**Integer(4)))

ValueError: chr() arg not in range(0x110000)
```

0.3. La llista dels caràcters de *SageMath*: de caràcters a codis

La funció ord( ) de *SageMath* proporciona el valor (decimal) del codi d'un caràcter. En cas de voler els codis d'un text, cal fer-ho caràcter a caràcter. Per tant, caldrà fer un petit programa.

In [12]:
```
1 llista=[ord(i) for i in "Llista de caràcters."]
2 print(llista)
```

```
[76, 108, 105, 115, 116, 97, 32, 100, 101, 32, 99, 97, 114, 224, 99
, 116, 101, 114, 115, 46]
```

Notem que el caràcter de codi més petit és l'espai (de codi 32), i que hi ha algun caràcter que no és ASCII (de codi 224), i que correspon a una vocal minúscula accentuada, à.

In [13]:
```
1 print([min(llista), max(llista)])
```

```
[32, 224]
```

Veiem quins codis numèrics tenen (en *SageMath*) les vocals accentuades:

In [14]:
```
1 vocals="ÀÁÂÄÈÉÊËÌÍÎÏÒÓÔÖÙÚÛÜàáâäèéêëìíîïòóôöùúûü"
```

In [15]:
```
1 cc=[ord(v) for v in vocals]
2 print(cc)
```

```
[192, 193, 194, 196, 200, 201, 202, 203, 204, 205, 206, 207, 210, 2
11, 212, 214, 217, 218, 219, 220, 224, 225, 226, 228, 232, 233, 234
, 235, 236, 237, 238, 239, 242, 243, 244, 246, 249, 250, 251, 252]
```

In [16]:
```
1 print([min(cc), max(cc)])
```

```
[192, 252]
```

Notem que les (nostres) vocals accentuades no esgoten la llista de codis (per exemple, quins caràcters corresponen al codis 195, 197, 198, 199, etcètera?. Fem la llista de tots els caràcters amb aquests codis:

In [17]:
```
ll=""
for v in range(192,253):
    ll=ll+chr(v)
print(ll)
```

```
ÀÁÂÃÄÅÆÇÈÉÊËÌÍÎÏÐÑÒÓÔÕÖ×ØÙÚÛÜÝÞßàáâãäåæçèéêëìíîïðñòóôõö÷øùúûü
```

0.4. Les funcions Codis(text) i Caracters(codis)

Es tracta de definir una funció Codis(text) que, aplicada a una tira de caràcters, text, en proporcioni la llista de codis numèrics.

I també, definir una funció inversa, Caracters(codis) que, aplicada a una llista de codis en proporcioni la tira de caràcters de la qual prové.

Observació. A fi que la llista sigui de codis, els elements de la llista han de ser del rang $0 \leq codi < m := 1114112$; però, en algun dels usos posteriors, la llista podria contenir valors fora d'aquest rang. A fi d'estalviar-nos feina i no fer una programació més feixuga (hauríem de comprovar que els codis són correctes i proporcionar, si cal, missatges d'error), ens limitarem a prendre el residu de la divisió per m de cadascun dels elements de la llista.

0.4.0. La funció Codis(text)

In [18]:
```
def Codis(text):
    return [ord(i) for i in text]
```

0.4.1. La funció Caracters(codis)

In [19]:
```
def Caracters(codis):
    m=1114112
    cars=""
    for i in range(len(codis)):
        cars=cars+chr(codis[i]%m)
    return cars
```

0.4.2. Exemples

Es tracta de comprovar en alguns exemples que les funcions són inverses l'una de l'altra (per a valors del rang $0 \leq codi < m$).

0.4.2.0.

In [20]:
```
text="Un text de prova."
print(text)
```

```
Un text de prova.
```

In [21]:
```
codis=Codis(text)
print(codis)
```

```
[85, 110, 32, 116, 101, 120, 116, 32, 100, 101, 32, 112, 114, 111,
118, 97, 46]
```

In [22]:
```
retext=Caracters(codis)
print(retext)
```

```
Un text de prova.
```

In [23]:
```
text==retext
```

Out[23]: True

Notem, a continuació, que per a valors fora del rang, les funcions no són (no ho poden ser) inverses l'una de l'altra: la composició Codis(Caracters(lta1)) no és la identitat sobre lta1.

In [24]:
```
lta1=[65,65+1114112]
print(lta1)
```

```
[65, 1114177]
```

In [25]:
```
txt1=Caracters(lta1)
print(txt1)
```

```
AA
```

In [26]:
```
lta2=Codis(txt1)
print(lta2)
```

```
[65, 65]
```

In [27]:
```
lta2==lta1
```

Out[27]: False

In [28]:
```
Codis(Caracters(lta1))==lta1
```

Out[28]: False

0.4.2.1.

Descodifiquem la llista següent de caràcters: [69, 102, 101, 99, 116, 105, 118, 97, 109, 101, 110, 116, 44, 32, 110, 111, 109, 233, 115, 32, 99, 97, 108, 105, 97, 32, 97, 112, 108, 105, 99, 97, 114, 45, 108, 105, 32, 108, 97, 32, 102, 117, 110, 99, 105, 243, 32, 67, 97, 114, 97, 99, 116, 101, 114, 115, 40, 32, 41].

In [29]:
```
lta=[69, 102, 101, 99, 116, 105, 118, 97, 109, 101, 110, 116, 44
```

In [30]:
```
Caracters(lta)
```

Out[30]: 'Efectivament, només calia aplicar-li la funció Caracters( )'

Fi del capítol 0