Foundations of Data Science for Biologists

# Working with text in Unix

BIO 724D

2025-JAN-28

Instructors: Paul Magwene and Greg Wray

# Regular expressions: quick review part 1

Any character: **.**

```
grep 'ca.e' Voyage.txt
```

finds line with `cane`, `case`, etc.

Any character from a set: [xyz]

```
grep 'c[au]p' Voyage.txt
grep 'c[^au]p' Voyage.txt
```

finds line with `cap`, `cup`, but not `copy`

negation: finds `copy`, but not `cap`, `cup`

Alternate strings: (x|y)

```
grep -E 'Ta|Va' Voyage.txt
grep -E 's(mo|al)' Voyage.txt
```

finds lines with `East`, `Zappo`, etc.

finds lines with `smooth`, `salmon`, etc.

# Regular expressions: quick review part 2

Anchors: ^ and $

```
grep '^Tahi' Voyage.txt
grep 'case$' Voyage.txt
```

finds lines that start with `Tahiti`, etc.

finds lines that end with `case`

Quantifiers: ? and * and + and {n} and {mn}

not reviewing today, but useful!

# Encodings

Computers store 0s and 1s; **encodings** specify how to store human-readable information

**ASCII** — first widespread encoding and basis for most later ones

- Name is an acronym for American Standard Code for Information Interchange
- Based on 7 bits, specifies 128 distinct printing and non-printing characters
- E.g., `1001000` indicates `<tab>` and `1100100` indicates `A`

**Unicode** — family of encodings used by internet and nearly every current computer

- First 128 characters identical to ASCII (for backward compatibility)
- Based on 1-4 bytes, potentially specifying >1,100,000 characters
- **UTF-8** is most widespread encoding by far; specifies >100,000 characters
- Includes mathematical symbols, arrows, scripts for many languages, emoji, etc.

# ASCII encoding

| $b_7 b_6 b_5$ → / Bits | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|
| $b_4$ $b_3$ $b_2$ $b_1$ / Column → / Row ↓ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 0 0 0 | 0 | NUL | DLE | SP | 0 | @ | P | ` | p |
| 0 0 0 1 | 1 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 0 0 1 0 | 2 | STX | DC2 | " | 2 | B | R | b | r |
| 0 0 1 1 | 3 | ETX | DC3 | # | 3 | C | S | c | s |
| 0 1 0 0 | 4 | EOT | DC4 | $ | 4 | D | T | d | t |
| 0 1 0 1 | 5 | ENQ | NAK | % | 5 | E | U | e | u |
| 0 1 1 0 | 6 | ACK | SYN | & | 6 | F | V | f | v |
| 0 1 1 1 | 7 | BEL | ETB | ' | 7 | G | W | g | w |
| 1 0 0 0 | 8 | BS | CAN | ( | 8 | H | X | h | x |
| 1 0 0 1 | 9 | HT | EM | ) | 9 | I | Y | i | y |
| 1 0 1 0 | 10 | LF | SUB | * | : | J | Z | j | z |
| 1 0 1 1 | 11 | VT | ESC | + | ; | K | [ | k | { |
| 1 1 0 0 | 12 | FF | FS | , | < | L | \ | l | \| |
| 1 1 0 1 | 13 | CR | GS | - | = | M | ] | m | } |
| 1 1 1 0 | 14 | SO | RS | . | > | N | ^ | n | ~ |
| 1 1 1 1 | 15 | SI | US | / | ? | O | _ | o | DEL |

sort order:

<tab>

<return>

<space>

symbols/punctuation

numerals

more symbols/punct.

upper case letters

more symbols

lower case letters

more symbols

US-ASCII 1967, the most widely adopted encoding during the early days of computers

# UTF-8 encodings

Notation

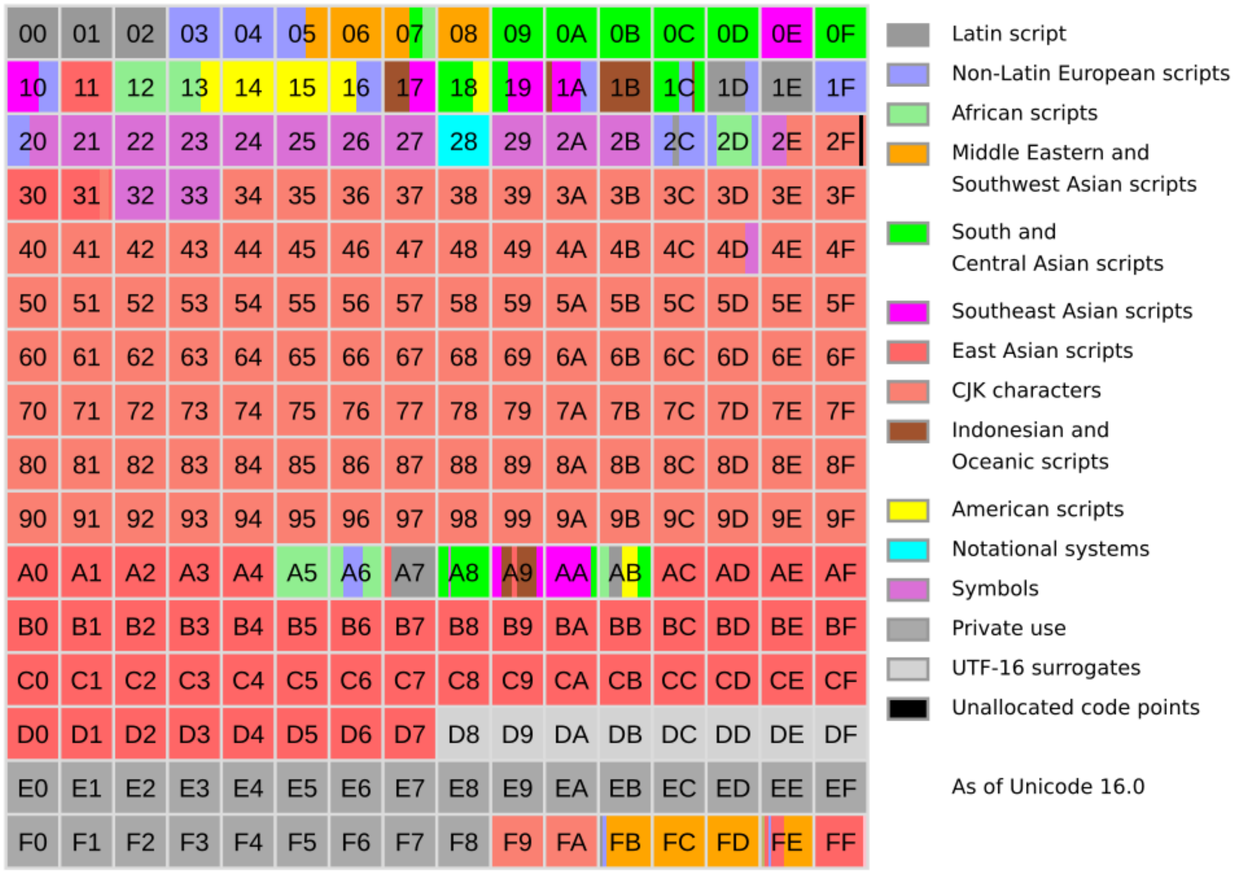
- **U+** followed by 4 or 6 **hexadecimal** numerals (left pad if needed)
- Values in range `U+0000-U+10FFFF` (vast majority of values are not assigned)
- First 2 digits define the **plane**; plane `0` does not require first 2 digits (`00`)
- Each plane contains 65,646 potential character encodings (**code points**)

Examples

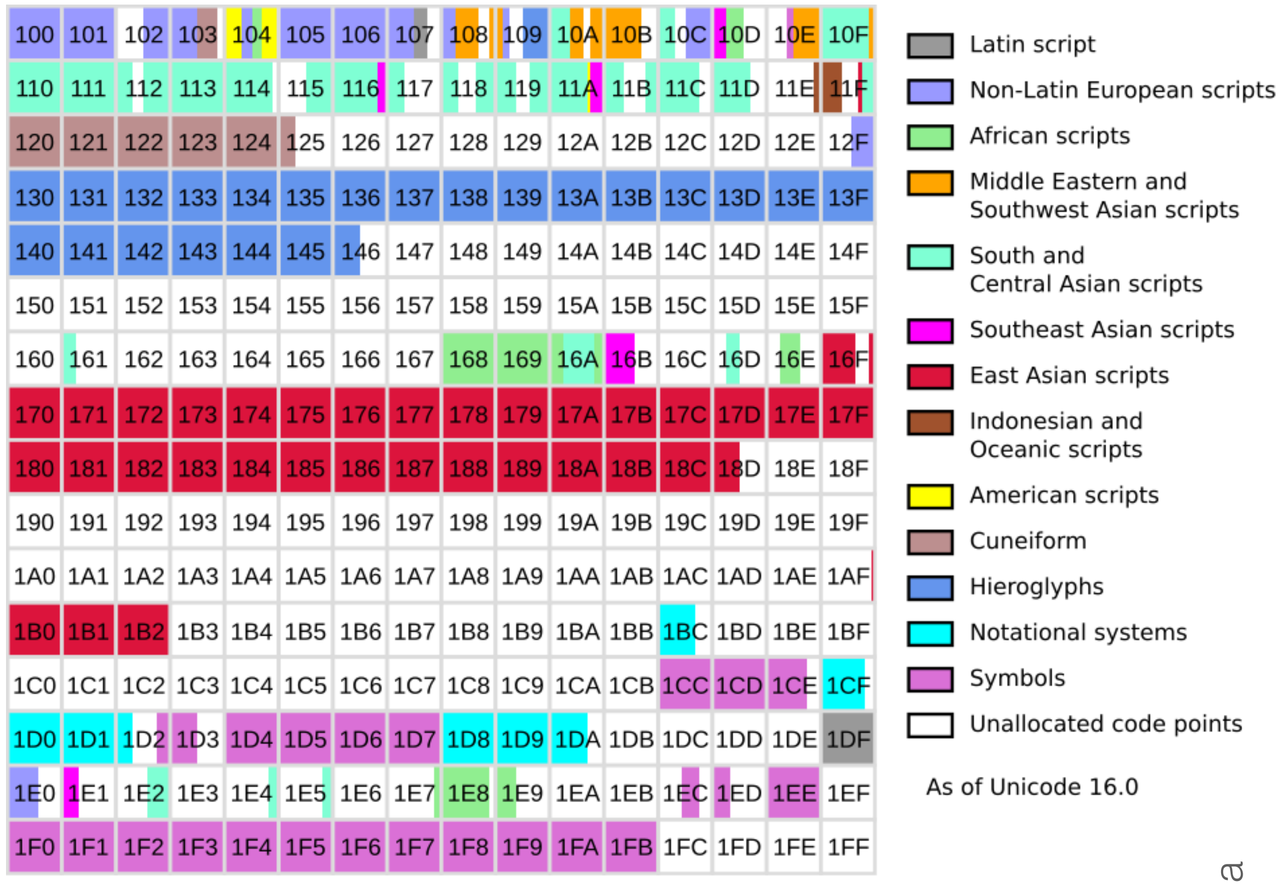
| | | |
|---|---|---|
| `U+004B` | K | Latin lower-case letter K |
| `U+006B` | k | Latin lower-case letter K |
| `U+0915` | क | Devanagari letter “kuh” |
| `U+0643` | ك | Arabic letter “kaf” |
| `U+20AC` | € | Euro currency symbol |
| `U+01F60E` | 😎 | smiling face with sunglasses (plane 1; requires 6 digits) |

# By far the most commonly used UTF-8 planes

## Plane 0




## Plane 1




Basic Multilingual Plane

Supplementary Multilingual Plane

Wikipedia

# Unicode links

## Official Unicode materials

Unicode v16.0 character code charts: http://www.unicode.org/charts/

Unicode standard: https://www.unicode.org/versions/Unicode16.0.0/

## Useful third-party resources

Shapecatcher: https://shapecatcher.com/ (search by shape)

Unicode Lookup: https://unicodelookup.com/ (search by word, character, or code)

EmNudge: https://unicode.emnudge.dev/ (search by word or code)

r21a's converter: https://r12a.github.io/app-conversion/ (type in character, get code)

Wikipedia: https://en.wikipedia.org/wiki/List_of_Unicode_characters (browse)