Awk: a brief intro

Jean-Baka Domelevo Entfellner
3rd-gen Genomics & Bioinformatics CoP
June 2021

Awk’s main features

- Processes one or more input files **line by line** (one input line = one *record*)
- A record is made of *fields* (field separator to be specified)
- The script tells Awk which actions to perform on which records, based on **filtering rules**
- Actions are delimited by curly braces (**{...}**). Code inside them: C-like syntax.
- Awk automatically maintains (updates) some internal variables, e.g. NR, NF, FS, OFS...

Awk’s processing loop

(1) Process actions associated with optional **BEGIN** rule

(2) Get the next record from current input file

(3) Scan sequentially all filtering rules, executing the actions corresponding to the selection patterns matching the current record, ignoring the other actions

(4) Loop back to (2) until the end of the input file is reached

(5) Process actions associated with optional **END** rule

Awk’s built-in variables

- **NR**: total number of records read so far
- **FNR**: total number of records read so far *in the current input file*
- **NF**: number of fields in the current record
- **FS**: field separator in use (default: whitespace sequences)
- **$0**: the whole current record (can be altered in place)
- **$1**, **$2**, … , **$(NF-1)**, **$NF**: the individual fields in the current record

Some selection patterns

- **NR == 2**: filters out the second record only
- **NR % 2 == 0**: even records (every second line)
- **NF > 0**: non-empty records (lines) only
- **/jb/**: records containing the pattern “jb”
- **$2 ~ /^jb/**: records whose second field starts with “jb”
- **flag == 1 && NF > 4**: combining boolean filters

Some built-in functions

- **print()**: to output (default action is **print $0**)
- **length()**: number of characters in a string
- **substr($0,2,5)**: substring of $0 of length 5 starting with its second character
- **index($3, “toto”)**: 1-base index for the first occurrence of “toto” in $3. Returns 0 if no occurrence found
- **exit**: stop processing the input file(s) and jump to the END rule (if it exists), then terminate