

# Unix Tools

Stanley Yao  
Computer Science Department  
University of Arizona

## awk – text processing



- Records (usually “line”): awk process the input text stream record by record.
- Fields (usually “word”): within a record, files are identified by \$1, \$2, ... and \$0 refers to the entire record
- Pattern-action statement
  - Pattern {actions}
  - Default pattern matches all records
  - Default action is copying the record onto the stdout
  - Special pattern BEGIN and END

Csc352-Summer03, Stanley Yao

2

## awk – Regular Expression

.	Any character
*	Zero or more of the preceding character
+	One or more of the preceding character
?	Zero or one of the preceding character
	Or
()	Grouping
[ ]	Character class (e.g. [a-z]). If the first character is '^', it means not.
^	Beginning of record
\$	End of record

Csc352-Summer03, Stanley Yao

3

## awk - Variables

- No need to declare
- Special variables
  - NR: record number of the current record
  - NF: number of fields in the current record
  - FILENAME: name of the current input file
  - \$1, \$2, ...: field 1, file2, ...
  - \$0: the current record

Csc352-Summer03, Stanley Yao

4

## awk - Examples

- who | awk '/mike/ {printf("%s\n", \$1);}'
- who | awk '\$1~/mike/ {printf("%s\n", \$1);}'
- who | awk '\$1~/yao/'
  
- ps | awk '\$1~/[0-9]/ {print \$1}'
- who | awk '{for (i=1; i<=NF; i=i+1) printf("%s, ", \$i); printf"\n"}'
- who | awk '{printf("%d: %s\n", NR, \$0);}'

Csc352-Summer03, Stanley Yao

5

## Further Readings

- Shelley Powers, Jerry Peek, Tim O'Reilly, Mike Loukides, *Unix Power Tools, Third Edition*, O'Reilly & Associates, 2002



Csc352-Summer03, Stanley Yao

6



## Acknowledgement

- John H. Hartman, *Classnotes for Csc352-Spring03*, CS Dept., University of Arizona, 2003