

LING 408/508: Computational Techniques for Linguists

Lecture 13

Administrivia

- Homework 6 review
- Regular expressions (regex) and awk

Homework 6

- Write awk code to:
 1. print a table of and **calculate the total percentage of population** for the top 10, 20 and 30 surnames
 2. read and print out the table with table headings **aligned** with the field values (use printf)

Rank	Name	Approximate percentage
1	González	4.79
2	Rodríguez	4.64
3	Hernández	4.01
4	Pérez	3.35
5	García	3.25

Homework 6 Review

- Basic solution:

```
awk  
'BEGIN {f="%5s %-14s %s\n"; printf f, "Rank", "Name", "%"}  
NR <= 10 {printf f, $1, $2, $3; s+=$3}  
END {print "Total %:", s}'  
surnames.txt
```

- Notes:

• NR	variable	Number of records (lines)
• f	variable	Format string
%s	format control	print as a string
%Ns	format control	print as a string in width <i>N</i> characters
%-Ns	format control	print as a string in width <i>N</i> characters, left-justified

Homework 6 Review

```
sandiway@sandiway-VirtualBox:~/Desktop$ awk 'BEGIN {f="%5s %-14s %s\n"; printf f,"Rank","Name","%"} NR <= 10 {printf f, $1, $2, $3; s+=$3} END {print "Total % : ", s}' surnames.txt
```

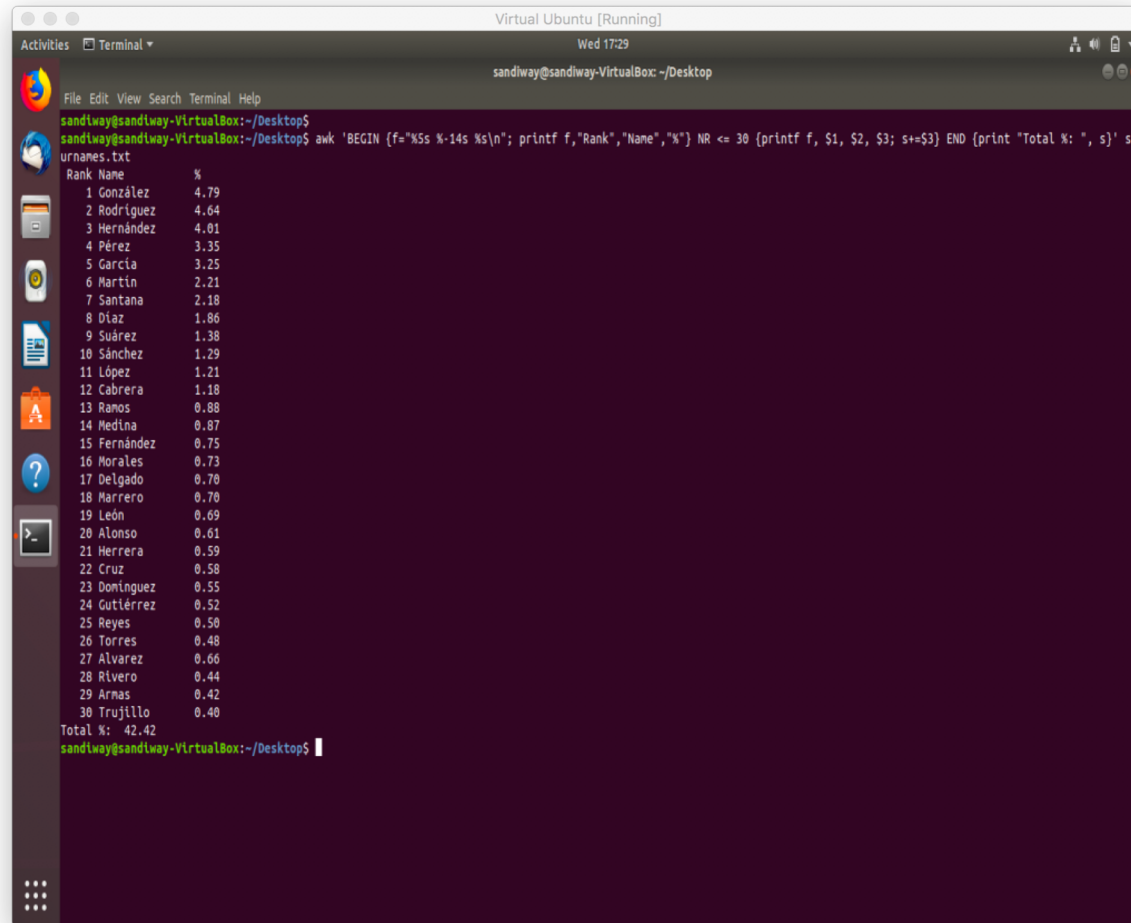
Rank	Name	%
1	González	4.79
2	Rodríguez	4.64
3	Hernández	4.01
4	Pérez	3.35
5	García	3.25
6	Martín	2.21
7	Santana	2.18
8	Díaz	1.86
9	Suárez	1.38
10	Sánchez	1.29

```
Total %: 28.96  
sandiway@sandiway-VirtualBox:~/Desktop$
```

Homework 6 Review

```
sandiway@sandiway-VirtualBox:~/Desktop$ awk 'BEGIN {f="%5s %-14s %s\n"; printf
f,"Rank","Name","%"} NR <= 20 {printf f, $1, $2, $3; s+=$3} END {print "Total %
: ", s}' surnames.txt
Rank Name          %
  1 González        4.79
  2 Rodríguez       4.64
  3 Hernández       4.01
  4 Pérez           3.35
  5 García          3.25
  6 Martín          2.21
  7 Santana         2.18
  8 Díaz            1.86
  9 Suárez          1.38
 10 Sánchez         1.29
 11 López           1.21
 12 Cabrera         1.18
 13 Ramos           0.88
 14 Medina          0.87
 15 Fernández       0.75
 16 Morales         0.73
 17 Delgado         0.70
 18 Marrero         0.70
 19 León            0.69
 20 Alonso          0.61
Total %: 37.28
sandiway@sandiway-VirtualBox:~/Desktop$
```

Homework 6 Review



The screenshot shows a terminal window titled "Virtual Ubuntu [Running]" with the date "Wed 17:29" and the user "sandway@sandway-VirtualBox: ~/Desktop". The terminal displays the following command and output:

```
sandway@sandway-VirtualBox:~/Desktop$  
sandway@sandway-VirtualBox:~/Desktop$ awk 'BEGIN {f="Xs %-14s %s\n"; printf f,"Rank", "Name", "X"} NR <= 30 {printf f, $1, $2, $3; s+=$3} END {print "Total X: ", s}' surnames.txt
```

Rank	Name	X
1	González	4.79
2	Rodríguez	4.64
3	Hernández	4.01
4	Pérez	3.35
5	García	3.25
6	Martín	2.21
7	Santana	2.18
8	Díaz	1.86
9	Suárez	1.38
10	Sánchez	1.29
11	López	1.21
12	Cabrera	1.18
13	Ramos	0.88
14	Medina	0.87
15	Fernández	0.75
16	Morales	0.73
17	Delgado	0.70
18	Marrero	0.70
19	León	0.69
20	Alonso	0.61
21	Herrera	0.59
22	Cruz	0.58
23	Domínguez	0.55
24	Gutiérrez	0.52
25	Reyes	0.50
26	Torres	0.48
27	Alvarez	0.66
28	Rivero	0.44
29	Arnas	0.42
30	Trujillo	0.40

Total X: 42.42

```
sandway@sandway-VirtualBox:~/Desktop$
```

Homework 6 Review

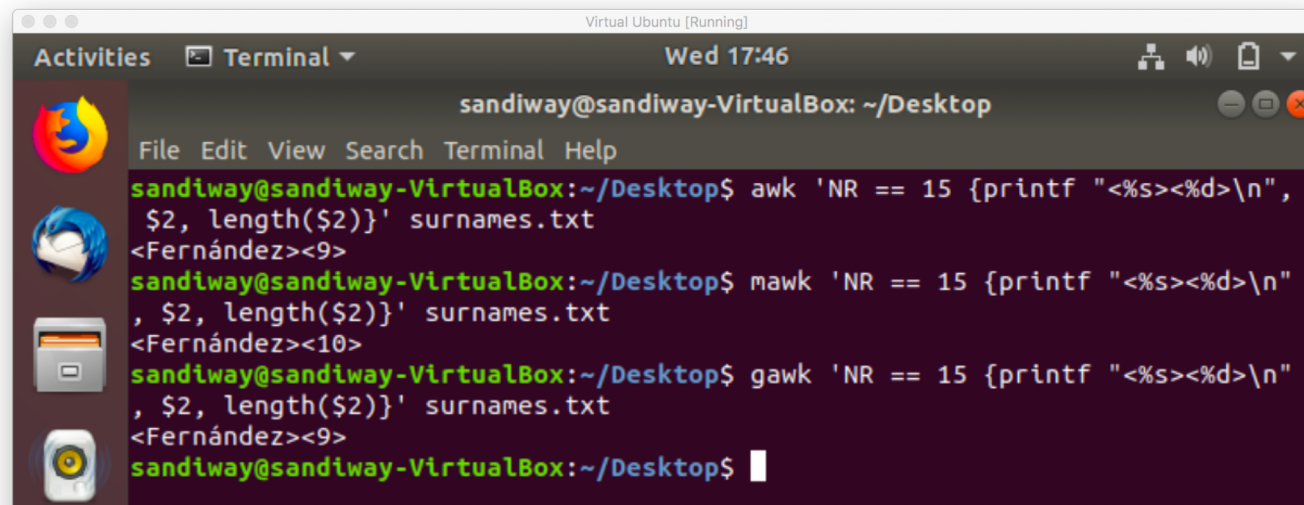
- Some of you may have run into a problem with `printf`'s behavior with `%s` (string printing) and UTF-8
- Note the `printf` tutorial is from gnu
- There are variants of `awk`: e.g. `mawk`, `gawk`

Homework 6 Review

```
Virtual Ubuntu [Running]
Wed 17:41
sandilway@sandilway-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
sandilway@sandilway-VirtualBox:~/Desktop$ awk 'BEGIN {f="%5s %-14s %s\n"; printf f,"Rank","Name","%"} NR <= 10 {printf f, $1, $2, $3; s+=$3} END {print "Total %: ", s}' surnames.txt
Rank Name %
1 González 4.79
2 Rodríguez 4.64
3 Hernández 4.01
4 Pérez 3.35
5 García 3.25
6 Martín 2.21
7 Santana 2.18
8 Díaz 1.86
9 Suárez 1.38
10 Sánchez 1.29
Total %: 28.96
sandilway@sandilway-VirtualBox:~/Desktop$ mawk 'BEGIN {f="%5s %-14s %s\n"; printf f,"Rank","Name","%"} NR <= 10 {printf f, $1, $2, $3; s+=$3} END {print "Total %: ", s}' surnames.txt
Rank Name %
1 González 4.79
2 Rodríguez 4.64
3 Hernández 4.01
4 Pérez 3.35
5 García 3.25
6 Martín 2.21
7 Santana 2.18
8 Díaz 1.86
9 Suárez 1.38
10 Sánchez 1.29
Total %: 28.96
```

```
Virtual Ubuntu [Running]
Wed 17:42
sandilway@sandilway-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
sandilway@sandilway-VirtualBox:~/Desktop$ mawk 'BEGIN {f="%5s %-14s %s\n"; printf f,"Rank","Name","%"} NR <= 10 {printf f, $1, $2, $3; s+=$3} END {print "Total %: ", s}' surnames.txt
Rank Name %
1 González 4.79
2 Rodríguez 4.64
3 Hernández 4.01
4 Pérez 3.35
5 García 3.25
6 Martín 2.21
7 Santana 2.18
8 Díaz 1.86
9 Suárez 1.38
10 Sánchez 1.29
Total %: 28.96
sandilway@sandilway-VirtualBox:~/Desktop$ gawk 'BEGIN {f="%5s %-14s %s\n"; printf f,"Rank","Name","%"} NR <= 10 {printf f, $1, $2, $3; s+=$3} END {print "Total %: ", s}' surnames.txt
Rank Name %
1 González 4.79
2 Rodríguez 4.64
3 Hernández 4.01
4 Pérez 3.35
5 García 3.25
6 Martín 2.21
7 Santana 2.18
8 Díaz 1.86
9 Suárez 1.38
10 Sánchez 1.29
Total %: 28.96
```

Homework 6 Review



```
Virtual Ubuntu [Running]
Activities Terminal Wed 17:46
sandivay@sandivay-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
sandivay@sandivay-VirtualBox:~/Desktop$ awk 'NR == 15 {printf "<%=s><%=d>\n",
, $2, length($2)}' surnames.txt
<Fernández><9>
sandivay@sandivay-VirtualBox:~/Desktop$ mawk 'NR == 15 {printf "<%=s><%=d>\n"
, $2, length($2)}' surnames.txt
<Fernández><10>
sandivay@sandivay-VirtualBox:~/Desktop$ gawk 'NR == 15 {printf "<%=s><%=d>\n"
, $2, length($2)}' surnames.txt
<Fernández><9>
sandivay@sandivay-VirtualBox:~/Desktop$
```

- Why? Reason is some printf implementations count **bytes** not (multibyte) characters
- cannot be fixed by changing locale to es_ES.UTF-8

awk: regex

- Reading assignment (**READ! READ! READ!**):
 - https://www.gnu.org/software/gawk/manual/html_node/Regexp.html
- Reference manual:
 - <http://manpages.ubuntu.com/manpages/bionic/en/man7/regex.7.html>

A pattern-action statement has the form

```
pattern { action }
```

pattern can be a regex
/regex/

A missing { action } means print the line; a missing pattern always matches. Pattern-action statements are separated by newlines or semicolons.

awk: regex

Helpful Links

Plans

Interacting with Others

Sign Up For UAlert!

UA Alert is a service that allows registered users – including University of Arizona students, faculty and staff – to receive emergency alerts on mobile phones or other mobile devices during a campus emergency.

[Read more](#)

UAlert

Call 911 for any emergency requiring police, fire or medical assistance.

Other Campus Resources

RESOURCE	PHONE NUMBER
University of Arizona Police Department (UAPD) ▼	520-621-8273
Risk Management and Safety ▼	520-621-1790
Office of Radiation, Chemical and Biological Safety ▼	520-626-6850

```
1 RESOURCE    PHONE NUMBER
2 University of Arizona Police Department (UAPD)  520-621-8273
3 Risk Management and Safety  520-621-1790
4 Office of Radiation, Chemical and Biological Safety  520-626-6850
5 Arizona Institutional Biosafety Committee  520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office  520-621-7057
8 Facilities Management  520-621-3000
9 Arizona Poison and Drug Information Center  800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

uanumbers.txt

awk: regex

1	RESOURCE	PHONE NUMBER
2	University of Arizona Police Department (UAPD)	520-621-8273
3	Risk Management and Safety	520-621-1790
4	Office of Radiation, Chemical and Biological Safety	520-626-6850
5	Arizona Institutional Biosafety Committee	520-621-5279
6	Campus Health Service	520-621-6490
7	Dean of Students Office	520-621-7057
8	Facilities Management	520-621-3000
9	Arizona Poison and Drug Information Center	800-222-1222
10	Recorded updates during campus emergencies	520-626-1222 (Tucson) 800-362-0101 (Toll free)

Task: print every line that has a toll-free number

```
awk '/800-/ {print $0}' uanumbers.txt
```

Arizona Poison and Drug Information Center 800-222-1222

Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)

awk: regex

```
1 RESOURCE      PHONE NUMBER¶
2 University of Arizona Police Department (UAPD)  520-621-8273¶
3 Risk Management and Safety  520-621-1790¶
4 Office of Radiation, Chemical and Biological Safety  520-626-6850¶
5 Arizona Institutional Biosafety Committee  520-621-5279¶
6 Campus Health Service 520-621-6490¶
7 Dean of Students Office  520-621-7057¶
8 Facilities Management  520-621-3000¶
9 Arizona Poison and Drug Information Center  800-222-1222¶
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)¶
```

Task: print every line that uses the term "safety"

```
awk '/[Ss]afety/ {print $0}' unumbers.txt
```

Risk Management and **Safety** 520-621-1790

Office of Radiation, Chemical and Biological **Safety** 520-626-6850

Arizona Institutional **Biosafety** Committee 520-621-5279

[Ss] means match either S or s
([^Ss] would be match any character other than S or s)

awk: regex

- Two useful string functions:
 - https://www.gnu.org/software/gawk/manual/html_node/String-Functions.html
 - `match(string, regexp [, array])`
 - Search *string* for the longest, leftmost substring matched by the regular expression *regexp* and return the character position (index) at which that substring begins (one, if it starts at the beginning of *string*). If no match is found, return zero.
 - The `match()` function sets the predefined variable **RSTART** to the index. It also sets the predefined variable **RLENGTH** to the length in characters of the matched substring. If no match is found, **RSTART** is set to zero, and **RLENGTH** to -1.
 - `substr(string, start [, length])`
 - Return a *length*-character-long substring of *string*, starting at character number *start*. The first character of a string is character number one.
 - If *length* is not present, `substr()` returns the whole suffix of *string* that begins at character number *start*.

awk: regex

A tab (\t) divides the two fields

```
1 RESOURCE    PHONE NUMBER
2 University of Arizona Police Department (UAPD) 520-621-8273
3 Risk Management and Safety 520-621-1790
4 Office of Radiation, Chemical and Biological Safety 520-626-6850
5 Arizona Institutional Biosafety Committee 520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office 520-621-7057
8 Facilities Management 520-621-3000
9 Arizona Poison and Drug Information Center 800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

```
awk 'match($0,/\t/) {print substr($0,1,RSTART)}' uanumbers.txt
```

RESOURCE

University of Arizona Police Department (UAPD)

Risk Management and Safety

Office of Radiation, Chemical and Biological Safety

Arizona Institutional Biosafety Committee

Campus Health Service

Dean of Students Office

Facilities Management

Arizona Poison and Drug Information Center

Recorded updates during campus emergencies

A list of all the resources

awk: regex

```
1 RESOURCE      PHONE NUMBER¶
2 University of Arizona Police Department (UAPD)  520-621-8273¶
3 Risk Management and Safety  520-621-1790¶
4 Office of Radiation, Chemical and Biological Safety  520-626-6850¶
5 Arizona Institutional Biosafety Committee  520-621-5279¶
6 Campus Health Service 520-621-6490¶
7 Dean of Students Office  520-621-7057¶
8 Facilities Management  520-621-3000¶
9 Arizona Poison and Drug Information Center  800-222-1222¶
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)¶
```

```
awk 'match($0, /520-[0-9]+-[0-9]+/) {print substr($0,RSTART,RLENGTH)}' uanumbers.txt
```

```
520-621-8273
520-621-1790
520-626-6850
520-621-5279
520-621-6490
520-621-7057
520-621-3000
520-626-1222
```

A list of all the local phone numbers

[0-9]+ means one or digits from the range 0-9

awk: regex

```
1 RESOURCE    PHONE NUMBER
2 University of Arizona Police Department (UAPD)  520-621-8273
3 Risk Management and Safety  520-621-1790
4 Office of Radiation, Chemical and Biological Safety  520-626-6850
5 Arizona Institutional Biosafety Committee  520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office  520-621-7057
8 Facilities Management  520-621-3000
9 Arizona Poison and Drug Information Center  800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

FS (Field separator): \t

```
awk 'BEGIN {FS="\t"} {print $1}' uanumbers.txt
```

RESOURCE

University of Arizona Police Department (UAPD)

Risk Management and Safety

Office of Radiation, Chemical and Biological Safety

Arizona Institutional Biosafety Committee

Campus Health Service

Dean of Students Office

Facilities Management

Arizona Poison and Drug Information Center

Recorded updates during campus emergencies

awk: regex

```
1 RESOURCE    PHONE NUMBER
2 University of Arizona Police Department (UAPD)  520-621-8273
3 Risk Management and Safety  520-621-1790
4 Office of Radiation, Chemical and Biological Safety  520-626-6850
5 Arizona Institutional Biosafety Committee  520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office  520-621-7057
8 Facilities Management  520-621-3000
9 Arizona Poison and Drug Information Center  800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

FS (Field separator): \t

```
awk 'BEGIN {FS="\t"} {print $2}' uanumbers.txt
```

PHONE NUMBER

520-621-8273

520-621-1790

520-626-6850

520-621-5279

520-621-6490

520-621-7057

520-621-3000

800-222-1222

520-626-1222 (Tucson) 800-362-0101 (Toll free)

awk: regex

```
1 RESOURCE    PHONE NUMBER
2 University of Arizona Police Department (UAPD)  520-621-8273
3 Risk Management and Safety  520-621-1790
4 Office of Radiation, Chemical and Biological Safety  520-626-6850
5 Arizona Institutional Biosafety Committee  520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office  520-621-7057
8 Facilities Management  520-621-3000
9 Arizona Poison and Drug Information Center  800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

Find the local exchanges (to area code 520):

```
gawk 'match($0, /520-([0-9]+)/, arr) {print arr[1]}' unumbers.txt
```

```
621
621
626
621
621
621
621
621
626
```

gawk exclusive:
arr[0] = entire match
arr[1] = submatch of 1st set of (...)
and so on...

awk: regex

```
1 RESOURCE    PHONE NUMBER
2 University of Arizona Police Department (UAPD) 520-621-8273
3 Risk Management and Safety 520-621-1790
4 Office of Radiation, Chemical and Biological Safety 520-626-6850
5 Arizona Institutional Biosafety Committee 520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office 520-621-7057
8 Facilities Management 520-621-3000
9 Arizona Poison and Drug Information Center 800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

Find the local exchanges (to area code 520) without duplicates:

```
gawk 'match($0, /520-([0-9]+)/, arr) {xch[arr[1]]=1}
END {for (x in xch) { print x}}'
uanumbers.txt
```

621

626

xch is an array we use to store the local exchange codes
arr[1] will be the local exchange code (used as key)
for the associative array xch
1 (assigned) is just a dummy value
(Note: Python dict = associative array)

awk: regex

```
1 RESOURCE      PHONE NUMBER
2 University of Arizona Police Department (UAPD) 520-621-8273
3 Risk Management and Safety 520-621-1790
4 Office of Radiation, Chemical and Biological Safety 520-626-6850
5 Arizona Institutional Biosafety Committee 520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office 520-621-7057
8 Facilities Management 520-621-3000
9 Arizona Poison and Drug Information Center 800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

Build a table of all the words used:

```
awk 'NR!=1 {for (i=1; i<=NF; i++) {word[$i]+=1}}
END {for (x in word) { printf "%12s %d\n", x,
word[x]}}' uanumbers.txt | sort -k2 -n
```

NR!=1 (pattern) skip 1st line (!= means *not equal to*)

NF = number of fields on a line

word = associative array of frequencies

| = pipe (output of awk into sort)

sort -k2 -n = command to sort on field 2 numerically (-n)

not case insensitive

```
Arizona 3
and 3
of 3
Management 2
Safety 2
Office 2
Institutional 1
800-362-0101 1
800-222-1222 1
520-626-6850 1
...
520-621-1790 1
emergencies 1
...
during 1
campus 1
Police 1
Poison 1
Health 1
Center 1
Campus 1
...
```

awk: regex

```
1 RESOURCE      PHONE NUMBER
2 University of Arizona Police Department (UAPD) 520-621-8273
3 Risk Management and Safety 520-621-1790
4 Office of Radiation, Chemical and Biological Safety 520-626-6850
5 Arizona Institutional Biosafety Committee 520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office 520-621-7057
8 Facilities Management 520-621-3000
9 Arizona Poison and Drug Information Center 800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

Build a table of all the words used (case-insensitive):

```
awk 'NR!=1 {for (i=1; i<=NF; i++) {word[tolower($i)]+=1}}
END {for (x in word) { printf "%12s %d\n", x, word[x]}}'
uanumbers.txt | sort -k2 -nr
```

tolower(*string*)

Return a copy of *string*, with each uppercase character in the string replaced with its corresponding lowercase character. Nonalphabetic characters are left unchanged. For example, tolower("MiXeD cAsE 123") returns "mixed case 123".

https://www.gnu.org/software/gawk/manual/html_node/String-Functions.html

```
arizona 3
and 3
of 3
management 2
safety 2
office 2
campus 2
institutional 1
800-362-0101 1
...
520-621-1790 1
information 1
emergencies 1
university 1
...
biosafety 1
students 1
recorded 1
chemical 1
(tucson) 1
...
```

awk: regex

```
1 RESOURCE      PHONE NUMBER
2 University of Arizona Police Department (UAPD) 520-621-8273
3 Risk Management and Safety 520-621-1790
4 Office of Radiation, Chemical and Biological Safety 520-626-6850
5 Arizona Institutional Biosafety Committee 520-621-5279
6 Campus Health Service 520-621-6490
7 Dean of Students Office 520-621-7057
8 Facilities Management 520-621-3000
9 Arizona Poison and Drug Information Center 800-222-1222
10 Recorded updates during campus emergencies 520-626-1222 (Tucson) 800-362-0101 (Toll free)
```

Build a table of all the words used (no numbers, no punctuation):

```
gawk 'NR!=1 {for (i=1; i<=NF; i++) {gsub(/^[^A-Za-z]/, "", $i);
word[tolower($i)]+=1}} END {for (x in word) { printf "%12s
%d\n", x, word[x]}}' uanumbers.txt | sort -k2 -nr
```

`gsub(regex, replacement [, target])`

Search *target* for all of the longest, leftmost, *nonoverlapping* matching substrings it can find and replace them with *replacement*. The 'g' in `gsub()` stands for "global," which means replace everywhere.

https://www.gnu.org/software/gawk/manual/html_node/String-Functions.html

arizona 3
and 3
of 3
management 2
safety 2
office 2
campus 2
institutional 1
information 1
emergencies 1
university 1
facilities 1
department 1
biological 1
radiation 1
committee 1
biosafety 1
students 1
recorded 1
chemical 1
updates 1
service 1
tucson 1
police 1