

C01

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C01.sh < liczby.in
1998 99 27 10 7 5 3 2 1
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C01.sh
1 9987 1 762 0 76 76 25 3 87 4 3 765 3
9987 765 762 87 76 25 4 3 1 0
```

```
#!/bin/bash
# ./C01.sh < liczby.in
read numbers
numbers=$(echo $numbers | tr ' ' '\n')
sorted_numbers=$(echo "$numbers" | sort -n -r -u)
echo $sorted_numbers
```

Wejście: liczby w 1 linii oddzielone linią spacji

Wyjście: liczby wejścia wypisane malejąco, bez powtórzeń

Komenda sort sortuje liczby wejścia. -n – po wartościach numerycznych, -r -malejąco/reverse, -u – unikalne wartości

C02

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C02.sh < auta.in
431
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C02.sh
bmw,m3,245
audi,a4,180
fiat,126p,100
mercedes,S-class,260
260
```

```
#!/bin/bash

#linie mozna podac recznie lub z pliku auta komenda ./C02.sh < auta.in
max_predkosc=0
najszybszy_samochod=""

while IFS=, read -r marka model predkosc; do
    if (( $(echo "$predkosc > $max_predkosc") )); then
        max_predkosc="$predkosc"
        najszybszy_samochod="$marka $model"
    fi
done

# echo "$najszybszy_samochod"
echo $max_predkosc
```

Wejście: uporządkowane trójki o elementach oddzielonych przecinkami

Wyjście: prędkość najszybszego samochodu

Plik czytuje kolejne linie z pliku/standardowego wejścia i przypisuje elementy trójki do poszczególnych zmiennych. Sprawdza czy obecna trójka ma większą prędkość niż dotychczas najwyższa, po czym jeżeli warunek jest spełniony nadpisuje dane. Po przejściu pętli wypisuje najwyższą znaną prędkość.

C03

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C03.sh < linie.in
To powinna byc 123 linijka pliku.... ale można sprawdzic :))
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ |
```

```
#!/bin/bash
n=123
declare -a lines

while IFS= read -r line; do
    lines+=("$line")
done

echo "${lines[n-1]}"
```

Linie można też wprowadzić ręcznie zamiast z pliku, ale wpisywanie 123 linii po enterze mija się z sensem :)

Wejście: wiele linii... lub plik zawierający dużo linii

Wyjście: linii nr 123, czyli o indeksie 122

W pętli while wczytuje linie do tablicy lines, a po jej przejściu wypisuje żadaną linię

C04

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C04.sh < linie.in
10: Moim zdaniem to nie ma tak, że dobrze albo że nie dobrze.
11: Gdybym miał powiedzieć, co cenię w życiu najbardziej, powiedziałbym,
12: że ludzi. Ekhm... Ludzi, którzy podali mi pomocną dłoń, kiedy sobie nie radziłem,
13: kiedy byłem sam. I co ciekawe, to właśnie przypadkowe spotkania wpływają na nasze życie.
14: Chodzi o to, że kiedy wyznaje się pewne wartości, nawet pozornie uniwersalne, bywa, że nie
15: znajduje się zrozumienia, które by tak rzec, które pomaga się nam rozwijać. Ja miałem szczęście,
16: by tak rzec, ponieważ je znalazłem. I dziękuję życiu. Dziękuję mu, życie to śpiew, życie to taniec,
17: życie to miłość. Wielu ludzi pyta mnie o to samo, ale jak ty to robisz?, skąd czerpiesz tę radość?
18: A ja odpowiadam, że to proste, to umiowanie życia, to właśnie ono sprawia, że dzisiaj na przykład
19: buduję maszyny, a jutro... kto wie, dlaczego by nie, oddam się pracy społecznej i będę ot, choćby sadzić...
20: znaczy... marchew.
```

```
n=10
m=20
declare -a lines

while IFS= read -r line; do
    lines+=("$line")
done

for ((i = n-1; i < m; i++)); do
    echo "${lines[i]}"
done
```

Wejście: wiele linii... lub plik zawierający dużo linii

Wyjście: linie z zakresu 10-20

Sczytuje linie z pliku/standardowego wejścia aż nie natrafi na koniec pliku/wejścia, a następnie wyświetla linie z podanego zakresu

C05

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C05.sh < linie.in
10
```

```
#!/bin/bash

tmp_file=$(mktemp)

while IFS= read -r line; do
    echo "$line" >> "$tmp_file"
done

# echo "cz,sz count:"
grep -io "cz\|sz" "$tmp_file" | wc -l
rm $tmp_file
```

Wejście: plik/tekst zawierający polskie znaki

Wyjście: ten sam tekst, lecz zamiast polskich znaków występuje symbol 'X'

Sczytuje linie z pliku/standardowego wejścia aż nie napotkam na znak końca pliku, oraz wpisuje je do pliku tymczasowego. Komendą grep wyszukuje wzorca podanego wyrażeniem regularnym, czyli szukamy 'sz' i 'cz'. Flaga -i ignoruje wielkość liter, a flaga -o powoduje, że szukamy wyłącznie w pod słowach, a nie liczymy wyłącznie całe wyrazy. Wc -l zlicza ilość które wyda grep

C06

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C06.sh < pl.in > pl.out
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ cat pl.in
Na wyścigach wyścigowych wyścigówka wyścigowa wyścignęła wyścigówkę wyścigową numer sześć.
Wyimaginowana gżegżółka zgrzeszyła przeciw gżegżółkowemu Bożkowi Grzegorzowi
Ż Ą Ć
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ cat pl.out
Na wyXXcigach wyXXcigowych wyXXcigXXwka wyXXcigowa wyXXcignXXXXa wyXXcigXXwkXX wyXXcigowXX numer szeXXXX.
Wyimaginowana gXXegXXXXXXka zgrzeszyXXa przeciw gXXegXXXXXXkowemu BoXXkowi Grzegorzowi
XX XX XX
```

```
#!/bin/bash
tr -c 'A-Za-z0-9[:space:][:punct:]' 'X'
```

Wejście: plik/tekst zawierający polskie znaki

Wyjście: ten sam tekst, w którym polskie znaki zamieniono na X

tr z flagą zamienia znaki z podanego zakresu na znak 'X', flaga -c sprawia, że odwracamy zasady klasyfikacji, czyli zostawiamy wyłącznie znaki z podanego wyrażenia regularnego, a wszystkie inne zmieniamy na znak 'X'

C07

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C07.sh < pl.in
3 Ż Ą Ć
2 Wyimaginowana gzężółka zgrzeszyła przeciw gzężółkowemu Bożkowi Grzegorzowi
1 Na wyścigach wyścigowych wyścigówka wyścigowa wyścigneła wyścigówkę wyścigową numer sześć.
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C07.sh
linia pierwsza
kolejna linia
no i kolejna
a ta będzie ostatnia
4 a ta będzie ostatnia
3 no i kolejna
2 kolejna linia
1 linia pierwsza
```

```
#!/bin/bash
tmp_file=$(mktemp)
i=1

while IFS= read -r line; do
    echo "$i $line" >> "$tmp_file"
    ((i=i+1))
done

tac $tmp_file > odwrotnie.out
rm $tmp_file
```

Wejście: plik / tekst

Wyjście: linie z wejścia wraz z indeksami wejściowymi w odwróconej kolejności

Sczytuje linie wejścia do pliku tymczasowego dopisując każdej z nich indeks. Po przejściu pętli odwracam plik tymczasowy komendą tac i przekierowuje wyjście do pliku odwrotnie.out. Na końcu usuwam plik tymczasowy.

C08

```
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ cat interp.in
Moim zdaniem to nie ma tak, że dobrze albo że nie dobrze.
Gdybym miał powiedzieć, co cenię w życiu najbardziej, powiedziałbym,
że ludzi. Ekhm... Ludzi, którzy podali mi pomocną dłoń, kiedy sobie nie radziłem,
kiedy byłem sam. I co ciekawe, to właśnie przypadkowe spotkania wpływają na nasze życie.
Chodzi o to, że kiedy wyznaje się pewne wartości, nawet pozornie uniwersalne, bywa, że nie
znajduje się zrozumienia, które by tak rzec, które pomaga się nam rozwijać. Ja miałem szczęście,
by tak rzec, ponieważ je znalazłem. I dziękuję życiu. Dziękuję mu, życie to śpiew, życie to taniec,
życie to miłość. Wielu ludzi pyta mnie o to samo, ale jak ty to robisz?, skąd czerpiesz tę radość?
A ja odpowiadam, że to proste, to umiowanie życia, to właśnie ono sprawia, że dzisiaj na przykład
buduję maszyny, a jutro... kto wie, dlaczego by nie, oddam się pracy społecznej i będę ot, choćby sadzić...
znaczy... marchew.
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ ./C08.sh
151
s473634@WSL:/mnt/c/Users/admin/Desktop/Studia_rok3/NLP/lab03$ cat interp.out
Moim zdaniem to nie ma tak że dobrze albo że nie dobrze
Gdybym miał powiedzieć co cenię w życiu najbardziej powiedziałbym
że ludzi Ekhm Ludzi którzy podali mi pomocną dłoń kiedy sobie nie radziłem
kiedy byłem sam I co ciekawe to właśnie przypadkowe spotkania wpływają na nasze życie
Chodzi o to że kiedy wyznaje się pewne wartości nawet pozornie uniwersalne bywa że nie
znajduje się zrozumienia które by tak rzec które pomaga się nam rozwijać Ja miałem szczęście
by tak rzec ponieważ je znalazłem I dziękuję życiu Dziękuję mu życie to śpiew życie to taniec
życie to miłość Wielu ludzi pyta mnie o to samo ale jak ty to robisz skąd czerpiesz tę radość
A ja odpowiadam że to proste to umiowanie życia to właśnie ono sprawia że dzisiaj na przykład
buduję maszyny a jutro kto wie dlaczego by nie oddam się pracy społecznej i będę ot choćby sadzić
znaczy marchew
```

```
#!/bin/bash
tr -d '[:punct:]' < interp.in > interp.out
wc -w < interp.out
```

Wejście: plik zawierający tekst z licznymi znakami interpunkcyjnymi

Wyjście: ten sam tekst, pozbawiony znaków interpunkcyjnych, liczba słów w tekście bez znaków interpunkcyjnych

Polecenie `tr -d` usuwa wszystkie znaki pasujące do podanego wyrażenia regularnego, w tym przypadku wszystkie znaki interpunkcyjne. Wyjście przekierowuje do pliku `interp.out` po czym wywołuje na nim polecenie `wc -w` które zlicza ilość słów.

W razie problemów z archiwum plików, wszystkie skrypty i pliki wejściowe są dostępne na githubie:

https://github.com/milosz-amg/Natural_language_processing/tree/master/lab03