

COMP2041 Week 6 Tutorial

- show echon.pl, test1.pl, test2.pl, test3.pl, test4.pl

Question 1

```
$total = 0;
$last = "";

while ($line = <STDIN>) {
    next if $line !~ /^(\\d+)\\s*(\\.+)\\s*$/;
    $count = $1;
    $species = $2;

    if ($species ne $last) {
        print "$total $last\\n" if $last;
        $total = 0;
        $last = $species;
    }

    $total += $count;
}

print "$total $last\\n" if $last;
```

Question 2

- if the filename ends with a '|', the filename is interpreted as a command that pipes output to us

```
foreach $url (@ARGV) {
    open F, "wget -q -O- $url|" or die;
    while ($line = <F>) {
        foreach $number (split(/[\\d\\- ]/, $line)) {
            $number =~ s/\\D//g;
            print "$number\\n" if length $number >= 8 && length $number <= 15;
        }
    }
}
```

Question 3

- a. (((x))) b. hell c. 11 12 13 (note that 10..20 is inclusive)

Question 4

- glob returns a list of files matching the pattern

```

$total = 0;
# or *.ch for the question
foreach $f (glob '*.pl,txt') {
    open F, "<$f" or die;
    @arr = <F>;
    printf "%7d %s\n", scalar @arr, $f;
    $total += @arr;
    close F;
}

printf "%7d total\n", $total;

```

Question 5

```

($n, $m, $file) = @ARGV;
open INPUT, "<", $file or die;

while (<INPUT>) {
    chomp;
    @chars = split //;
    # do some checking on n and m if more than characters
    print @chars[$n-1..$m-1], "\n";
}

close INPUT;

```

```
./cut.pl 1 4 q1.txt
```

Question 6

```

while (<>) {
    chomp;
    ($id, $mark) = split; # default split is space
    if ($mark !~ /^(d{1,2}|100)$/) {
        $grade = "??";
    } else {
        $grade = "FL";
        $grade = "PS" if $mark >= 50 && $mark <= 100;
    }
    $extra = ($grade ne "??") ? "" : " ($mark)";
    print "$id $mark$extra\n";
}

```

Question 7

```

%phones = ();
open(PHONES, "<phones.txt") or die;

```

```
while(<PHONES>) {
    chomp;
    ($id, $type, $number) = split ',';
    $phones{"$id:$type"} = $number;
}
close PHONES;

open(PEOPLE, "<people.txt") or die;
while(<PEOPLE>) {
    chomp;
    ($id, $name, $address, $suburb) = split ',';
    print "$name\n$address, $suburb\nPhones: ";
    @numbers = ();

    foreach $type ("mobile", "home", "work") {
        push @numbers, $phones{"$id:$type"}." ($type)"
            if defined $phones{"$id:$type"};
    }
    print @numbers > 0 ? join(", ", @numbers) : "?", "\n\n";
}
close PHONES;
```