Revision Exercises with Perl, routing, switching and LDAP

1 Perl

1. Write a regular expression to match two or more "**x**"s followed by one or more "**y**"s followed by any number of "**z**"s.

Solution: All three of these are correct:

```
/xx+y+z*/
/xxx*yy*z*/
/x{2,}y{1,}z{0,}/
```

2. Write a regular expression to match any number of backslashes "\" followed by any number of asterisks "∗". Note that "any number" might be zero.

Solution: /***/

```
MSRP6604R1    Student Progression Rank Details            Page 1 of 4
Academic Year : 2002/03           Course Board Control No.          APS1238


Progression Course : 95551/2    in HW
                     95570/2    in LW
                     95573/2    in LW
                     95731/2    in LW
Code          Student No.     Name                  Attained  Loc Course/YearClass
95551/2       987654321       WAN Siu Ming (970)        41.0  TY      41230/1G
95551/2       976543210       FUNG Siu Ming (618)       42.1  TY      41456/2B
95551/2       965432109       YUET Siu Ming (639)       43.2  TY      41789/3B
95551/2       954321098       WONG Siu Ming, Joe (571)  44.3  TY      41012/1G
95551/2       943210987       KLIE Siu Ming (486)       45.4  TY      41345/2G
95551/2       932109876       WANG Siu Hing (370)       46.5  TY      41678/3G
95551/2       921098765       WONG Siu Ming (436)       47.6  TY      41901/1G
95551/2       910987654       FONG Siu Ming (359)             TY      41234/3G
95570/2       909876543       HO Siu Ming (133)         49.8  TY      41567/2B
```

Figure 1: A text file containing data

3. Figure 1 shows a text file containing data. Write a Perl program to print all student numbers.

Solution:
```
while ( <> ) {
        print "$1\n" if /(\d{9})/;
}
```

4. Write a perl program to print the course, year and class, each in a separate column, separated by a tab character.

Solution:
```
while ( <> ) {
        print "$1\t$2\t$3\n" if /\s(\d{5})\/(\d)([a-zA-Z])/;
}
```

5. Write a perl program to generate a password for each student according to the following algorithm:

   - Take the student's name (but without the number in brackets);
   - Remove all spaces and non-alphabetical characters from the name;
   - Use the first eight characters left, append them to the last four digits of the student number.

   Your program should print:

   - the student number;
   - the full name of the student;
   - the student's generated password

   all separated by a single tab character. There should be one line of output per student.

Solution: Here is one of many possible solutions. Here I have tried to be clear rather than very concise.

```
#! /usr/bin/perl -w

use strict;

while ( <> ) {
    if ( /(\d{9})\s+(\S.*)\s\(/ )
    {
        my ( $student_id, $name ) = ( $1, $2 );
        $student_id =~ /(\d{4})$/;
        my $pw_first = $1;
        my $pw_second = $name;
        $pw_second =~ s/[^a-zA-Z]//g;
        $pw_second =~ /^(.{8})/;
        $pw_second = $1;
        my $passwd = "$pw_first$pw_second";
        print "$student_id\t$name\t$passwd\n";
    }
}
```

Now some people were confused by the pattern match binding operator, =∼, and by the substitute operator, s/../../. Here is a little program that gives some examples of both:

```perl
#! /usr/bin/perl -w

use strict;

# Normally a match is made against $_.
# If you want to match against another string, use the binding
# operator, =~
# Here are some examples:

my $string = 'abc';
$string =~ /.(.)./;
my $what_is_this = $1;
# Notice that no change happened in $string:
print "\$string=$string, \$what_is_this=$what_is_this\n";
# see what the substitution operator does:
$string =~ s/b/B/;
print "\$string=$string\n";
# with no g modifier:
$string =~ s/./Z/;
print "\$string=$string\n";
# Now see what adding the g modifier does:
$string =~ s/./X/g;
print "\$string=$string\n";
# $ ./show-match
# $string=abc, $what_is_this=b
# $string=aBc
# $string=ZBc
# $string=XXX
```

2   Routing and Switching

1. Before a set of candidate network routes can be aggregated, they must be in the _____, and the routes in binary format must have the _____.

Solution: Before a set of candidate network routes can be aggregated, they must be in the <u>same network</u>, and the routes in binary format must have the <u>same prefix</u>.

2. Figure 2 on the next page shows a network with 5 routers and 10 subnets. You may select IP addresses from the block of addresses 172.12.0.0/19. You must leave at least one quarter of these addresses available for other purposes. The requirements are that each of subnets 1, 2,..., 8 must support up to 130 computers, while subnets 9 and 10 must each support up to 600 computers.

   (a) Allocate a suitable block of addresses to each of the ten subnets that will allow maximum route aggregation.

Solution: General strategy: determine the lower and upper limits on each subnet.
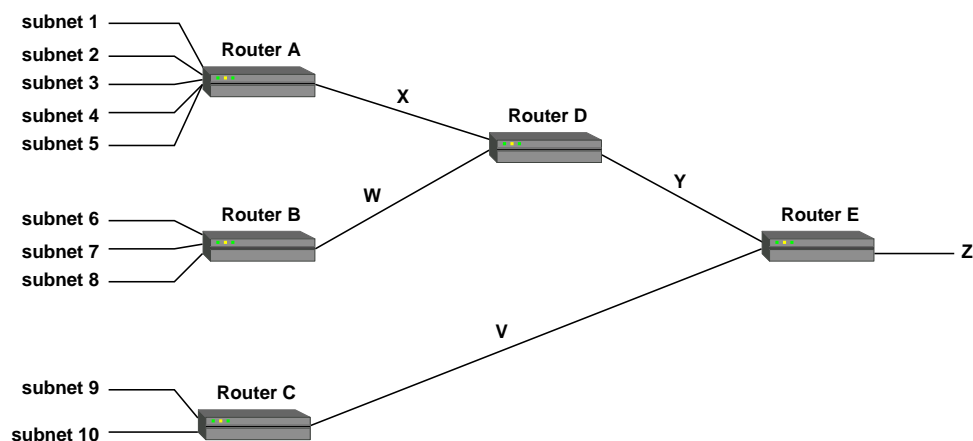
Figure 2: A network with five routers and ten subnets.

The minimum size of each of the first 8 subnets is 256, i.e., a /24 subnet, as 2^8 is the lowest power of 2 that contains 130.

Let us allocate the lowest 8 /24 blocks:

subnet	network
subnet 1	172.12.0.0/24
subnet 2	172.12.1.0/24
subnet 3	172.12.2.0/24
subnet 4	172.12.3.0/24
subnet 5	172.12.4.0/24
subnet 6	172.12.5.0/24
subnet 7	172.12.6.0/24
subnet 8	172.12.7.0/24

The minimum size of subnets 9 and 10 is $2^{10} = 1024$, since $2^{\lceil \log_2 600 \rceil} = 2^{10}$, giving a subnet size of /22.

Let us allocate the next lowest 2 /22 blocks:

subnet	network
subnet 9	172.12.8.0/22
subnet 10	172.12.12.0/22

This would use only half of the available addresses.

(b) Given your selection in the previous part, with *no* route summarisation active on the routers, list the routes that would be advertised by router A at X, by router B at W, by router C at V, by router D at Y, and by router E at Z.

Solution: *Without* route summarisation, router A will advertise one route for each of its five subnets:

172.12.0.0/24

172.12.1.0/24

172.12.2.0/24

172.12.3.0/24

172.12.4.0/24

and router B will advertise one route for each of *its* three subnets:

172.12.5.0/24

172.12.6.0/24

172.12.7.0/24

Router C will advertise one route for each of its two subnets:

172.12.8.0/22

172.12.12.0/22

Router D will advertise one route for each of the eight subnets behind routers A and B:

172.12.0.0/24

172.12.1.0/24

172.12.2.0/24

172.12.3.0/24

172.12.4.0/24

172.12.5.0/24

172.12.6.0/24

172.12.7.0/24

Finally router E will advertise one route for each of the ten subnets behind routers A, B and C:

172.12.0.0/24

172.12.1.0/24

172.12.2.0/24

172.12.3.0/24

172.12.4.0/24

172.12.5.0/24

172.12.6.0/24

172.12.7.0/24

172.12.8.0/22

172.12.12.0/22

(c) What would be a necessary requirement for the routers to support route aggregation?

Solution: The routers should be running a classless routing protocol, such as OSPF or RIP2.

(d) Repeat part 2b, but for the case where the routers all support route summarisation.

Solution: For router A, the first four routes will be aggregated (summarised), so there will be two routes advertised:

172.12.0.0/22

172.12.4.0/24

Router B will summarise the last two routes:

172.12.5.0/24

172.12.6.0/23

Router C will summarise both routes into one:

172.12.8.0/21

Router D will summarise the two routes from router A and the two routes from router B into one route:

172.12.0.0/21

Router E will aggregate the route from router C with the route from router D into one single route: 172.12.0.0/20

Note that here I have ignored the routing to any IP addresses used on the serial links on the routers.

3. When using portable notebook computers that should remain on one subnet, what type of VLAN should be employed? Are there any difficulties in supporting such a VLAN arrangement?

Solution: They should use a dynamic VLAN configuration. No matter what data port the notebook is plugged into, it will belong to one particular VLAN, and hence, in one subnet.

The main drawback of this system is maintaining the database of MAC address — VLAN mappings that must be stored in the switches that support this dynamic VLAN.
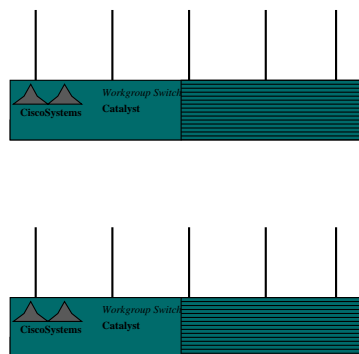
Figure 3: Two switches, each with five ports.

4. Two switches each have five ports, as shown in figure 3. Both switches are to support three VLANs, VLAN1, VLAN2 and VLAN3. Computers will be connected to each of these three VLANs on each of the two switches.

   (a) Each VLAN is to support a separate subnet. How would you connect the switches so that computers on each VLAN can communicate with each other?

   Support your answer with a diagram showing the connection of the switches. Label the VLANs. Indicate whether each port is a trunk port or an access port.

Solution: See figure 4 on the next page.
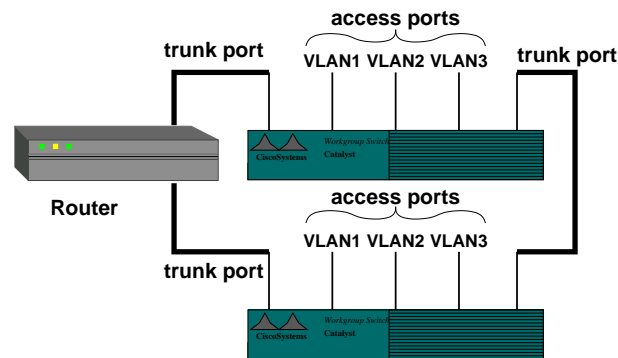
Figure 4: Two switches, each with five ports connected with a router. The trunk ports on the right are connected together to connect the VLANs on the two switches together. Each VLAN is a separate subnet. The router connects these subnets together.

(b) One additional port is added to each switch. Now two users with portable computers, one belonging to VLAN2 and the other belonging to VLAN3 want to be able to plug into the newly added port on either switch and remain in their respective VLAN without any manual configuration. How should you configure the new ports in each switch to facilitate this?

Solution: We would configure them as dynamic VLANs, and add one notebook's MAC address to dynamic VLAN2, the other to dynamic VLAN3.

If notebook-1 has MAC address MAC-1, and notebook-2 has MAC address MAC-2, and we want notebook-1 on VLAN2, and notebook-2 on VLAN3, then we must enter the following data into the switch's dynamic VLAN configuration:

MAC address	Which VLAN
MAC-1	VLAN2
MAC-2	VLAN3

3   Directories and LDAP

Many of you have not completed the tutorial sheet "LDAP Filters and Searching LDAP directories." I suggest that you complete it.