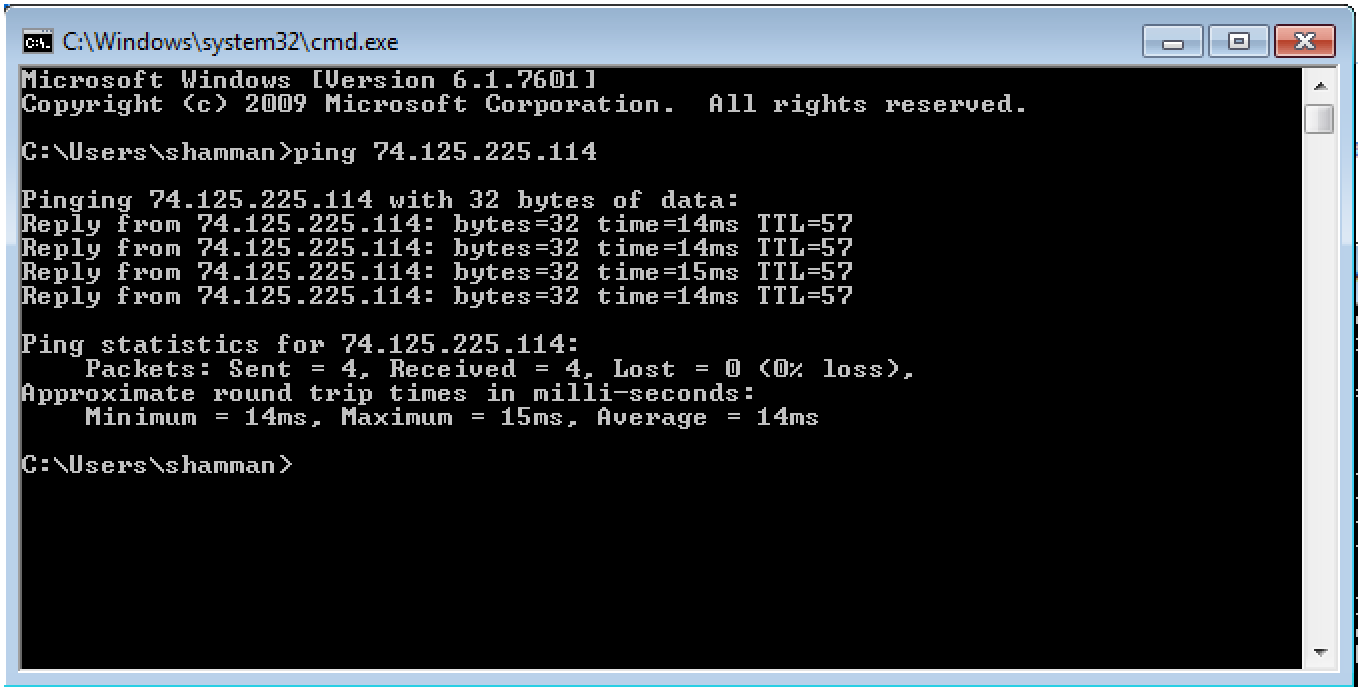
PA2

Description

Below are two screenshots of Window 7's ping program, issued from my computer's command prompt. The first one shows a successful ping (a Google IP address) and the second one shows a failed ping (a random unreachable IP address).



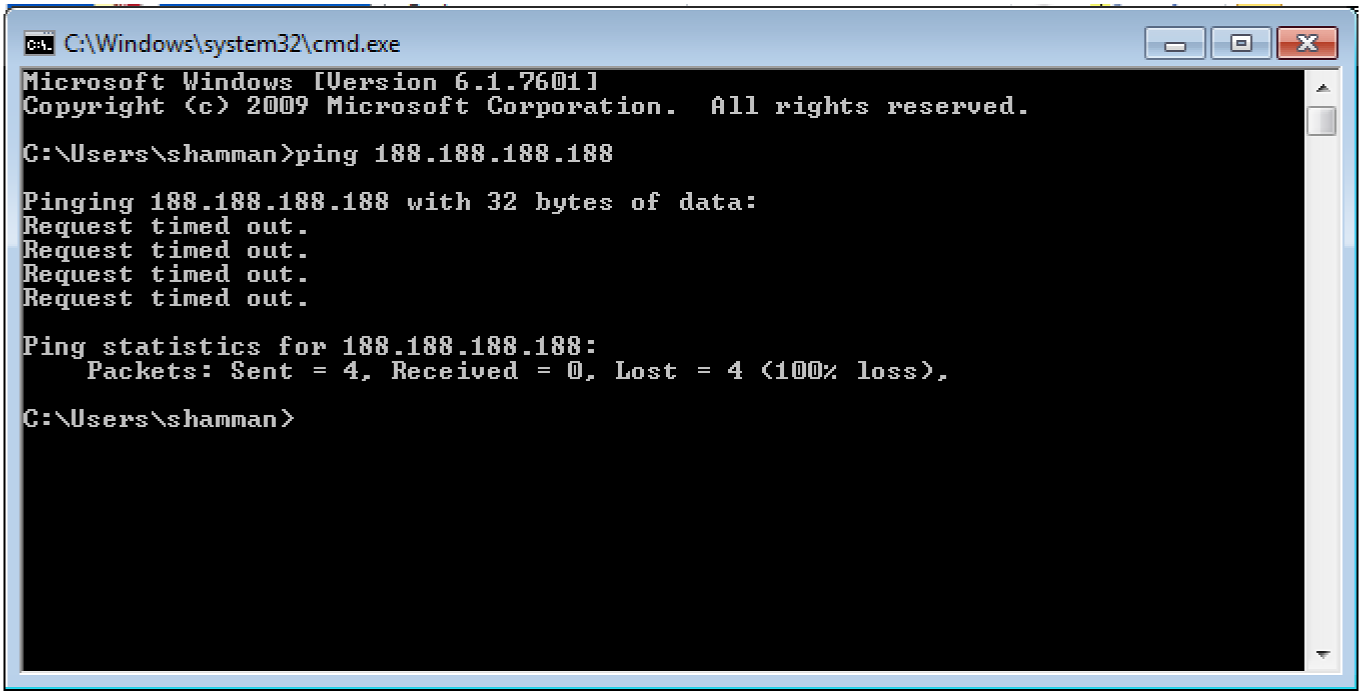
```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\shamman>ping 74.125.225.114

Pinging 74.125.225.114 with 32 bytes of data:
Reply from 74.125.225.114: bytes=32 time=14ms TTL=57
Reply from 74.125.225.114: bytes=32 time=14ms TTL=57
Reply from 74.125.225.114: bytes=32 time=15ms TTL=57
Reply from 74.125.225.114: bytes=32 time=14ms TTL=57

Ping statistics for 74.125.225.114:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 14ms, Maximum = 15ms, Average = 14ms

C:\Users\shamman>
```



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\shamman>ping 188.188.188.188

Pinging 188.188.188.188 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 188.188.188.188:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\Users\shamman>
```

The ping program measures RTT by sending an ICMP echo request packet (type 8, code 0) to the specified IP address and relies on the recipient to send an ICMP echo reply packet (type 0, code 0) back.

Your assignment is to write a program called JavaPing that mimics the behavior of ping. Because there is no way within Java to send and receive ICMP packets, you will instead implement ping the old fashioned way: by relying on the recipient to run the Echo Protocol (RFC 862 – a pleasantly short RFC). Echo echoes all the bytes that it receives directly back to the sender.

The Echo protocol has been superseded by ICMP and it is no longer in widespread use. It is turned off by default on most (all?) computers. However, I was able to locate a machine here on Cedarville's network that is for a limited time only running the service. You can find it at IP address: 163.11.238.205. **This particular server's Echo service is protected by a firewall and only accepts UDP packets from IP addresses in the range 163.11.0.0/16 and from port 7777.**

The only things that you need to know about the ping program can be deduced from examining its output (see screenshots above). The output from your implementation of ping should look **identical** to the actual ping, with one exception: you will not be able to report the TTL since that is a network layer field (it is in the IP packet header) and you have no access to that layer from within Java.

Note: you need to run JavaPing from a command prompt (just like the real ping) in order to receive full credit.

EXTRA CREDIT

You can earn up to 9 extra points by locating non-Cedarville, non-machines-you-control IP addresses that are running the Echo service. Each IP address found will result in 3 extra points, up to a maximum of 9 points.

WHAT TO TURN IN

At the beginning of class on the day it is due, hand in the following items in this order:

1. A screenshot of the console output of a successful ping to 163.11.238.205
2. A screenshot of the console output of a failed ping (you pick the IP address)
3. If you did the EC, a screenshot of the console output of a successful ping to a non-Cedarville, non-machine-you-control, IP address. One screenshot for each IP address you found.
4. The source code of your program.