# Simple Mail User Agent

## Goal

This programming lab includes two sections: i) use of the telnet to manually send an electronic mail through a SMTP mail server; ii) implementation of a python program to send an electronic mail.

## Part I: Sending email using Telnet

To make this step you must know the mail server of your mail domain. To obtain this information a query can be sent to the DNS server, having information about your mail domain, so that the MX resource record is retrieved. For example, the mail domain of ana.maria@gmail.com is gmail.com. The following command asks the DNS server for the servers responsible for managing the electronic mail in this domain:

```
> nslookup –type=MX gmail.com
```

Suppose now that the DNS server reply includes the gsmtp147.google.com mail server. The following command establishes a TCP connection from your system to port 25 at the server, the port address of SMTP:

```
> telnet gsmtp147.google.com 25
```

You may now insert the SMTP commands and inspect the answers of the server. The example presented below includes a sequence of commands to send an email from alberto to ana.maria@gmail.com. The string “alberto” is a name chosen by the sender and has no meaning. Some agents put the name of the computer. This name should have no special characters or spaces:

```
HELO alberto
MAIL FROM: <alberto@ualg.pt>
RCPT TO: <ana.maria@gmail.com>
DATA
SUBJECT: Hi
Hi Ana, Have you been studying a lot? Alberto.
.
QUIT
```

To complete this first step you should:
1) Send an electronic mail to yourself and check if it arrives to your mailbox.
2) Observe the header (extended) of the message and compare it with what you have sent. Write your conclusions, in particular where do the fields “From:”, “To:”, “RCPT TO:”, “Subject:”, etc, appear. Identify the header and body of the message.

## Part 2: Sending email using Python

You should write a program in Python that establishes a TCP connection with the mail server, through the socket interface, and sends an electronic mail. You must insert in your code all the details of the message you what to send. A possible skeleton for your code:

Prof. Dra. Noélia Correia, Universidade do Algarve, 2017.

```
from socket import *

# Mensagem a ser enviada
msg = '\r\nRedes de Computadores e fixe!'
endmsg = '\r\n.\r\n'

# Servidor de email
mailserver = 'smtp.ualg.pt'
portnumber = 25

# Criacao de socket e estabelecimento de conexao TCP com mailserver
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((mailserver, portnumber))

# Analise da primeira resposta do servidor

…

# Envio do comando HELO para servidor

...

# Envio do comando MAIL FROM para servidor

…

# Envio do comando RCPT TO para servidor

…

# Envio do comando DATA para servidor

...

# Envio da mensagem para servidor

…

# Envio do comando QUIT para servidor

...

# Fim do programa
```