

Redes de Computadores  
2019/2020

***Introdução ao Projecto:***  
***“RC Forum”***

*IST LEIC*

## 2 componentes:

- *Forum Server (FS)*
- *User Application*

user: vouga.ist.utl.pt

```
$/user -n 192.168.1.1 -p 58000
> reg 12345
  User "12345" registered
> tl
  available topics:
  1 - net (proposed by 11111)
  2 - wifi (proposed by 22222)
> ts 2
  selected topic: wifi (22222)
> ql
  available questions about wifi:
  1 - RTS
> qg 1
  stored files:
    wifi/RTS.txt, wifi/RTS_01.txt
  Q: What is RTS?
  A01: request to send
> as myRTS.txt
  answer submitted with success
> exit
$
```

### UDP

REG 12345\n  
RGR OK\n  
LTP\n  
LTR 2 net:11111 wifi:22222\n  
LQU wifi\n  
LQR 1 RTS:22222:1\n

### TCP

GQU wifi RTS\n  
QGR 33333 12 What is RTS? 0 1  
 1 44444 15 request to send 0\n  
ANS 12345 wifi RTS 68  
 RTS/CTS is used to reduce  
 collisions due to the  
 hidden node problem. 0\n  
ANR OK\n

FS: (TCP, UDP) tejo.ist.utl.pt:58000

```
$/FS -p 58000
user: 192.168.129.2 12345
List topics
List questions for topic: wifi
Send question files: wifi/RTS
New answer received: wifi/RTS
```

User command	User – FS (UDP)	User – FS (TCP)
<i>register userID</i> <i>reg userID</i>	REG→   ←RGR	
<i>topic_list</i> <i>TI</i>	LTP→   ←LTR	
<i>topic_select topic</i> <i>ts topic_number</i>	(local)	
<i>topic_propose topic</i> <i>tp topic</i>	PTP→   ←PTR	
<i>question_list</i> <i>ql</i>	LQU→   ←LQR	

User command	User – FS (UDP)	User – FS (TCP)
question_get question qg question_number		GQU→   ←QGR
question_submit question text_file [image_file.ext]  qs question text_file [image_file.ext]		QUS→   ←QUR
answer_submit text_file [image_file]  as text_file [image_file]		ANS→   ←ANR

# RC Forum

O código desenvolvido em C deve estar convenientemente **estruturado** e **comentado**.

As chamadas de sistema ***read()*** e ***write()*** podem ler e escrever, respetivamente, um numero de bytes inferior ao que lhes foi solicitado – deve garantir que ainda assim a sua implementação funciona corretamente.

Quer o processo cliente quer o processo servidor devem terminar graciosamente pelo menos nas seguintes situações de falha:

- **mensagens do protocolo erradas** vindas da entidade par correspondente;
- condições de **erro das chamadas de sistema**

O código a entregar: ficheiros fonte dos programas (*user*, *FS*), *Makefile*, e *ficheiros auxiliares*.

Entrega por e-mail, **até dia 18 de Outubro de 2019, às 23h59mn.**

Deve criar um único ficheiro de arquivo **zip** com todos os ficheiros fonte e outros ficheiros necessários à execução das aplicações. O arquivo deve estar preparado para ser aberto para o diretório corrente e compilado com o comando `make`. O nome do ficheiro submetido deve ter o seguinte formato: **proj<número do grupo>.zip**.