

Laboratórios de Comunicações III

MiECom (2º ano)

Projecto integrado

Ano Lectivo de 07/08

1 Objectivos

Com este projecto integrado pretende-se sedimentar os conhecimentos introduzidos nas aulas teóricas de Microprocessadores, Sistemas de Computação e Métodos/Paradigmas da Programação, relativos a:

- métodos rigorosos (orientados a objectos) de análise de problemas (requisitos) e desenvolvimento de software;
- métodos de programação imperativa (ou declarativa) suportados em algoritmos e estruturas de dados lineares e sua implementação em *Assembly* e *C/C++*.
- desenho e especificação de circuitos electrónicos baseados em microprocessador e respectiva interface entre as saídas/entradas digitais do microcontrolador e equipamento de entrada/saída;
- princípios básicos da comunicação entre computadores via porta série.

2 Pré-requisitos

Ter obtido aprovação em Laboratórios Integrados I e frequentado Laboratórios Integrados II.

3 Resultados da aprendizagem

No final do semestre, os alunos devem ter adquirido as seguintes competências.

Competências gerais :

- Utilizar correctamente o ambiente laboratorial, incluindo os equipamentos de medida básicos em electrónica;
- Utilizar ferramentas de desenvolvimento de software.

Competências específicas :

- Descrever a função de um microprocessador.
- Identificar os componentes que formam um sistema baseado num microprocessador (microprocessador, memória, interfaces de entrada e saída).
- Construir um sistema baseado num microprocessador que inclua o uso de displays, teclados e memórias.
- Estruturar um programa em Assembly dum microcontrolador.
- Dominar os fundamentos da programação em linguagem C, utilizar estruturas de dados em memória (arrays e listas ligadas simples), e manipular ficheiros (de texto ou binários) em acesso sequencial ou directo;
- Usar protocolos de transferência de dados: RS-232 (assíncrono) e PS/2.

4 Organização e Funcionamento

O projecto será desenvolvido em **grupos de 2 alunos** dentro e fora das aulas da disciplina (2 sessões semanais de 2 horas cada).

Em cada aula estarão presentes dois docentes que irão esclarecendo questões específicas dentro da sua área de trabalho.

Nos pontos de controlo (ver calendário abaixo) e no fim do semestre, cada grupo apresentará à equipe docente o trabalho realizado e os resultados obtidos, devendo entregar um relatório técnico de desenvolvimento devidamente estruturado e fundamentado, escrito em L^AT_EX (ver modelo disponível na página W3 da disciplina).

Em cada um desses pontos de controlo, serão escolhidos aleatoriamente alunos de alguns grupos para apresentar oralmente à turma (com recurso a um projector) o trabalho do grupo, de modo a promover uma discussão alargada das soluções encontradas.

4.1 Calendarização

O projecto deve ser executado ao longo de todo o semestre (o 1^o do 2^a ano), com 17 semanas de contacto¹, estando a *entrega final* agendada para a primeira semana após o fim das aulas (**4 a 9 Fev**).

Para controlo da situação e avaliação intermédia, haverá *3 apresentações intercalares* do projecto, nas semanas de: **(8-12out)**, **(5-9nov)** **(17-21dez)**.

As semanas de 29out-2nov, 19-23nov e 10-14dez serão integralmente dedicadas a *práticas de microprocessadores*.

Nota muito importante: Grupos que *falhem nos 2 primeiros pontos de avaliação intercalar*, por não comparecerem ou não atingirem uma avaliação positiva, *serão reprovados à disciplina*.

¹As aulas arranca na 2^a semana do período lectivo, a 24 de Setembro, e as 2 últimas semanas, a partir de 4 de Fevereiro, serão reservadas para a avaliação final do semestre.

5 Enunciado

Ma Urgência dos Hospitais usa-se hoje em dia um sistema de triagem dos doentes designado por **Triagem de Manchester (TM)**. De acordo com a TM, quando um doente entra é de imediato avaliado por uma equipe de para-médicos que lhe atribui uma cor conforme a gravidade do seu estado, gravidade essa que é aferida de acordo com a medição de um conjunto standard de parâmetros (que serão indicados posteriormente):

- vermelho – emergente (em risco de vida)
- laranja – muito urgente (situação grave, mas ainda não na hora da morte)
- amarelo – urgente
- verde – menos urgente
- azul – não urgente (não há risco, pode regressar ao Centro de Saúde)

Nesse contexto e sabendo dos projectos integrados que se realizam neste curso, solicitaram-lhes que desenvolvessem um **sistema para apoio à triagem e atendimento de doentes (SATAD)**.

O que se pretende é uma aplicação que:

- dê apoio aos para-médicos na fase de triagem,
- assista a enfermagem no encaminhamento dos doentes da sala de espera para a consulta,
- e, por fim, apoie o médico na consulta e no registo do diagnóstico a decisão final (fica em observação, é internado, tem alta).

Para apoio aos médicos e para-médicos, o SATAD mostra a ficha de identificação do doente ao qual estão associados todos os seus episódios hospitalares, sendo cada episódio descrito por três componentes:

- a ficha de avaliação do seu estado à chegada, com os tais parâmetros standard acima referidos (preenchida pelos para-médicos no acto da triagem);
- o diagnóstico e prescrição, textos corridos preenchidos pelo médico no acto da consulta;
- decisão final, um dos três valores possíveis acima enumerados, cujo preenchimento também é da responsabilidade do médico no fim do acto da consulta.

O sistema central, onde estão armazenadas as fichas de todos os doentes e respectivos episódios clínicos e ao qual acedem os para-médicos, é responsável por criar e manter uma Lista de Espera (LE), associando a cada uma das 5 cores o doente (identificado pelo número do seu cartão nacional de saúde) e a senha de atendimento atribuída por ordem absoluta de chegada.

Para isso o SATAD avalia automaticamente a cor do doente em função das medidas dos sinais vitais que constam da ficha de avaliação, de acordo com um conjunto de regras pré-estabelecidas (cada grupo deve definir as suas).

A componente do SATAD responsável localmente (na sala de espera) pela encaminhamento dos doentes para os gabinetes médicos onde são atendidos, afixa no painel de chamadas o número da senha e a cor do próximo doente a ser atendido e o número do gabinete para onde se deve dirigir.

Para isso, o controlador local recebe um sinal indicando que o médico de um dado gabinete (identificado pelo seu número) está livre e consulta o sistema central, o qual o informa da senha e código do doente de acordo com a LE criada dando prioridade aos de cor vermelha e por ai abaixo até ao amarelo. Nessa altura o doente seleccionado é retirado da LE.

No fim da consulta, o médico comunica (através de um teclado numérico, simplificado) ao sistema local o identificador do doente o diagnóstico (código) e a decisão final (código), para que o registo central desse episódio hospitalar seja actualizado.

5.1 Tarefas a Desenvolver

Para isso é necessário realizar as seguintes tarefas, algumas das quais devem prosseguir em paralelo:

1. fazer a análise exaustiva do problema no seu todo e de cada componente—*sistema de gestão central, sistema local*—de modo a conceber a arquitectura do sistema e representar essa arquitectura através de um diagrama de blocos.
2. criar um modelo orientado aos objectos (à semelhança do que foi feito no ano passado em **Java**) para descrever em detalhe cada componente; Para isso, desenhe o diagrama de classes e suas relações e indique à parte os atributos e métodos de cada classe.
3. desenvolver, no computador central, um programa em C/C++ para fazer a gestão dos doentes e da triagem, conforme se descreveu acima; esse programa deve ter uma interface simples (alfa-numérica tradicional) e deve trabalhar com os dados em memória—armazenadas em *arrays* e *listas-ligadas*, conforme for mais apropriado—gravando toda a informação num ficheiro binário no fim da execução para se poder recuperar numa execução seguinte.
numa fase final, poderá optimizar-se o programa usando uma interface gráfica (via janelas) e ficheiros de acesso directo².
4. desenvolver, localmente, a ligação (hardware e software) entre o microcontrolador o painel de chamadas (displays alfa-numéricos) e o teclado numérico do médico.
5. desenvolver o módulo de comunicação RS-232 entre o sistema central e o controlador.

6 Material necessário

Qt.	Descrição	Obs.
1	PC / Windows	com Visual C/C++ + WinEdt/MikTex
1	Micro-Controlador	Phillips 8951 com placa de desenvolvimento
1	Electrónica diversa	

²Tais melhorias serão consideradas na avaliação.

7 Elementos de estudo

Além dos *Manuais dos fabricantes* dos diversos equipamentos utilizados, sugere-se a consulta à página oficial da disciplina, em

www.di.uminho.pt/~prh/curECli307.html

para ver a lista de bibliografia recomendada.