

Entregue o código desenvolvido em linguagem C, devidamente indentado e comentado, e um relatório com a descrição das soluções. Inclua na entrega o *makefile* que permita gerar os ficheiros executáveis a partir do código fonte. O relatório deverá ser um guia para a compreensão do código desenvolvido e não uma mera tradução deste para língua natural. Contacte o docente se tiver dúvidas. Encoraja-se a discussão de problemas e soluções com colegas de outros grupos, mas recorda-se que a partilha directa de soluções leva, no mínimo, à anulação das entregas de todos os envolvidos.

O calendário de exames das unidades curriculares (UC) de um curso deve cumprir um conjunto de regras. Por exemplo: deve ser garantido um intervalo mínimo de dias entre as provas de primeira e segunda época de uma UC; as provas de duas UC do mesmo semestre devem ter entre si um intervalo mínimo de dias; se, numa das épocas, as provas de duas UC de semestres próximos coincidem, então na outra época essa coincidência não se deve repetir.

Para facilitar a validação destes mapas foi desenvolvido um programa em Java, cujas fontes se encontram em [anexo](#), que valida um calendário de exames segundo um conjunto configurável de regras. Algumas destas regras aplicam-se às datas de cada UC individualmente, enquanto que as outras se aplicam a cada par possível de UC. O programa recebe na linha de comando o nome de dois ficheiros de texto, um com informação sobre as UC do curso (*ProgramCourses*) e outro com o calendário a avaliar (*Exams*). Exemplos destes ficheiros também se encontram em [anexo](#).

Os ficheiros descritos têm uma linha para cada UC com elementos separados por uma barra vertical. Por exemplo: a linha “PI|B|16|AVE|RCp;LS” no primeiro ficheiro indica que PI é do tipo B (obrigatória), funciona apenas no 5º semestre (*bitmap* com bit a 1 nos semestres em que funciona), tem dependência forte de AVE e fraca de RCp e LS; a linha “PI|8|22” no segundo ficheiro indica que PI tem a primeira época no dia 8 do mapa e a segunda época no dia 22 do mapa (valores entre 0 e N).

1. Inspirando-se nas fontes do programa em Java, escreva um programa em C com os mesmos objectivos. A estrutura geral do programa em C deve seguir a do programa em Java, garantindo-se a mesma facilidade de alteração:
  - a) Deve existir, pelo menos, um módulo em C correspondente a cada módulo em Java.
  - b) O módulo para carregamento de informação (*DataLoader*) é usado para ler todos os ficheiros de texto.
  - c) As regras a aplicar são indicadas nos ficheiros “*validators1.txt*” e “*validators2.txt*”
  - d) A alteração, remoção ou adição de regras não deve obrigar a gerar uma nova versão do ficheiro executável que contém o programa de validação.
2. Acrescente um validador para garantir uma distância mínima *mi* entre UC de semestres consecutivos, excepto se forem dependências fortes, caso em que a distância mínima será de *md*, com  $md \leq mi$ .
3. Acrescente outro validador que faça sentido e que use a informação sobre as disciplinas serem ou não obrigatórias

Data limite de entrega: 11 de Janeiro de 2010

*Bom trabalho!*

NOTA: O calendário de exames do DEETC contém, neste semestre, marcações de provas para cerca de oito dezenas de UC em que aproximadamente metade delas funcionam em mais do que um dos cinco cursos do departamento (LEIC, LEETC, LERC, MEIC e MEET). Se pretender validar o calendário de exames real da LEIC no contexto do DEETC, contacte o docente.