Projeto de Laboratórios de Informática III (LI3)
Lic. Engª Inform. (LEI) - 2º ano - UMinho - 2012/2013

João M. Fernandes, João L. Sobral, F. Mário Martins

Linguagem C: Processamento estatístico de informação bibliográfica

1 Introdução

Este documento apresenta as linhas condutoras do primeiro projeto de LI3. O projeto deve ser codificado na linguagem C (com recurso ao `gcc`) e fazer uso dos conhecimentos adquiridos nas UCs de Programação Imperativa (PI), Algoritmos e Complexidade (AlgC) e Arquiteturas de Computadores (ArqC). Um dos objetivos deste projeto é permitir aos alunos exercitar as boas práticas e as técnicas que adquiriram nessas UCs num caso prático de média dimensão.
Valoriza-se na avaliação do trabalho que a construção do projeto obedeça a alguns princípios importantes, tais como:

- a escolha das estruturas de dados para representação e processamento da informação que o problema aborda;
- a utilização de algoritmos e definições apresentados em PI e AlgC;
- a escrita de código genérico, modular e reutilizável, que possa ser aproveitado noutros projetos futuros;
- a análise e otimização do desempenho da aplicação, com recurso a matéria abordada em ArqC.

O enunciado do projeto aborda um conjunto de requisitos cuja concretização em linguagem C requer a criação de estruturas de dados que foram estudadas previamente. Os alunos devem escolher as estruturas que pareçam mais adequadas (i.e., melhores segundo alguns critérios) e justificar no relatório, a apresentar na defesa do projeto, as razões e vantagens das escolhas efectuadas.

2 Requisitos nucleares

O **DBLP** (www.dblp.org/db; DBLP Computer Science Bibliography) é um website com fins não comerciais que permite armazenar e visualizar informação bibliográfica relativa a artigos científicos (em revistas e em conferências) na área da informática.
Cada entrada bibliográfica DBLP segue um formato bem definido e inclui, entre outros campos, um número, uma lista de autores, um título, o nome do artigo e o ano de publicação. O seguinte exemplo mostra uma concretização deste formato.

> 385 Björn Franke: Statistical Performance Modeling in Functional Instruction Set Simulators. ACM Trans. Embedded Comput. Syst. (TECS) 11(1):22-25 (2012)

Pretende-se desenvolver uma aplicação em C que permita fazer processamento estatístico da informação bibliográfica obtida no DBLP para os artigos publicados num conjunto de revistas e/ou conferências. Essa aplicação é constituída por um conjunto de módulos nucleares e um conjunto de módulos complementares, tal como indicado na figura 1. Estes módulos são descritos nas subsecções seguintes.

2.1 Módulo reconhecedor de entradas (Fase 1)

Este módulo (A) deverá processar informação armazenada em ficheiros textuais (B e C) e produzir um conjunto básico de estatísticas (D e E). A informação relativa a cada conferência ou revista é guardada num ficheiro textual (B). Os ficheiros de revistas devem ter um nome do tipo "`j-*.txt`" (e.g., `j-STTT.txt`) e os de conferências do tipo "`c-*.txt`" (e.g., `c-ACSD.txt`). Em cada linha desses ficheiros está uma entrada bibliográfica, de acordo com o formato definido no anexo A.
O ficheiro lista.txt (C) indica as conferências e revistas a considerar, bem como o número mínimo de páginas para que um artigo seja considerado uma entrada válida (i.e., não seja rejeitado/ignorado):

1. na 1ª linha, indica-se o número mínimo de páginas dos artigos a processar; artigos com número menor de páginas são rejeitados
2. em cada uma das linhas seguintes, é indicado o nome dum ficheiro relativo a uma conferência ("c-*.txt") ou revista ("j-*.txt") a considerar para efeitos de processamento.

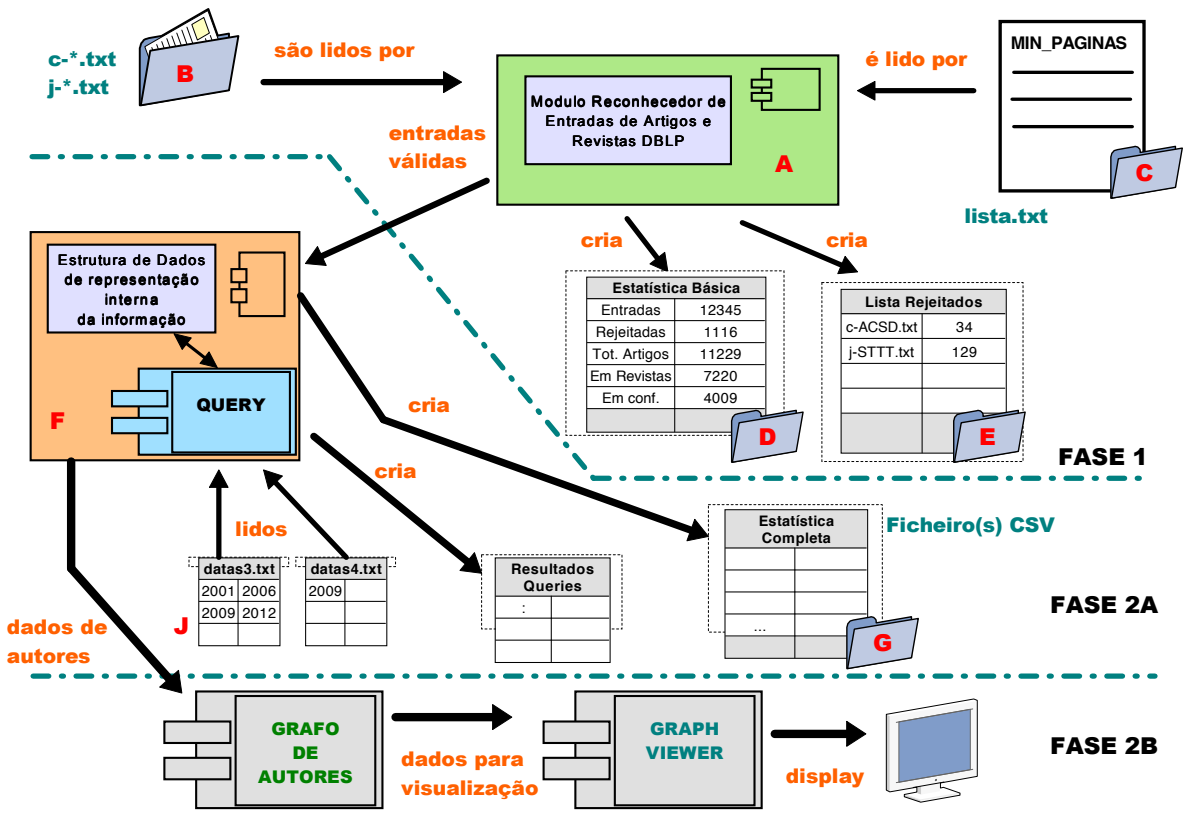

Figura 1: Arquitetura geral da aplicação

Este módulo (A) deverá produzir dois relatórios:

1. (D), guardado no ficheiro `D.txt`, com as seguintes estatísticas básicas:
   (a) número de entradas processadas
   (b) número total de entradas rejeitadas
   (c) número de entradas consideradas como artigos (não rejeitadas)
   (d) número de artigos em revistas (não rejeitadas)
   (e) número de artigos em conferências (não rejeitadas)

Exemplo:

```
Estatistica basica
------------------
12345 entradas
1116 rejeitadas
11229 artigos
  7220 em revista
  4009 em conferencia
```

2. (E), guardado no ficheiro `E.txt`, com a lista de entradas rejeitadas. Para cada ficheiro processado deverá indicar o nome do ficheiro e o número de entradas desse ficheiro que foram rejeitadas.

   Exemplo:

```
Lista Rejeitadas
----------------
c-ACSD.txt 34
c-STACS.txt 55
c-ICECS.txt 12
j-STTT.txt 129
```

Este módulo (A) também irá produzir informação que será utilizada por outros módulos da aplicação.

2.2 Módulo de representação e processamento da informação (Fase 2A)

Este módulo (F) deverá produzir um conjunto de informações relativas às entradas que foram consideradas válidas pelo módulo (A). O objetivo principal é contabilizar o número de autores de cada artigo, ou seja, não é necessário saber quais são os nomes dos autores dos artigos, mas apenas quantos eles são. Assim, pretende-se produzir um relatório (G) que mostra:

1. para cada ano, quantos artigos existem com os vários números de autores (ordenado decrescentemente);
2. número total de artigos por número de autores;
3. número total de artigos publicados para vários intervalos temporais (entre dois anos, inclusive); os intervalos de anos são indicados em (J; `datas3.txt`);
4. para um dado ano, indicado em (J; `datas4.txt`), a percentagem (com 2 casas decimais arredondadas) de artigos para cada número de autores.

O aplicação deve apresentar os resultados no ecrã e exportar essa informação para um ficheiro (G). Este ficheiro, de nome `G.csv`, deve obedecer ao formato CSV para poder ser posteriormente lido por programas que processam folhas de cálculo, e.g., MS-Excel.
O ficheiro (G) deve conter: o ano, o número de autores e o número de artigos reconhecidos com esse número de autores. Os campos devem ser separados por vírgulas e cada campo deve ser delimitado por aspas. O ficheiro (G) deve conter uma linha por cada entrada, incluindo uma linha inicial com a identificação dos campos, tal como no exemplo seguinte:

```
"ano","#autores","#artigos"
"1971","1","3"
...
"2011","1","38"
```

```
"2011","2","49"
"2011","3","75"
"2011","4","115"
"2012","1","224"
"2012","2","246"
"2012","3","246"
"#autores","#artigos"
"1","345"
"2","774"
"3","1311"
"4","653"
...
"intervalo","#artigos"
"2001-2006","3219"
"2009-2012","7609"
"ano","#autores","percentagem"
"2009","1","33.51"
"2009","2","22.60"
"2009","3","24.68"
"2009","4","12.12"
"2009","5","5.55"
"2009","6","1.54"
"2009","10","0.00"
```

3 Requisitos complementares (Fase 2B)

Existem ainda dois requisitos complementares que poderão ser implementados, de forma a permitir aos alunos atingir notas mais elevadas. Cada aluno terá que decidir quais os requisitos deste tipo (não obrigatórios) que pretende implementar. No caso de optar por incluir requisitos deste tipo, deverá incorporá-los na mesma aplicação que implementa os requisitos nucleares.

3.1 Otimização da aplicação

Devem ser efetuadas melhorias na aplicação visando a otimização do tempo necessário para a produção dos relatórios:

- **análise da escalabilidade da aplicação**. Pretende-se medir o tempo necessário para percorrer toda a estrutura de dados em função do número de entradas armazenadas. O aluno deverá também definir as entradas a inserir na estrutura de dados para que esta análise traduza condições reais.
- **implementação eficiente de pesquisas por número de artigos**. Pretende-se que as estruturas de dados suportem eficientemente pesquisas por número de autores, por exemplo, saber quantos artigos existem em cada ano para um dado número de autores.

3.2 Rede social de co-autores

Deve ser gerada informação que mostre para cada autor dos artigos processados, a lista de todos os seus co-autores (i.e., todos aqueles com quem partilha a co-autoria de pelo menos um artigo). Esta informação deve ser mostrada visualmente e corresponde a uma espécie de rede social dos autores, com as relações de autoria entre eles.

4 Relatório

O relatório a entregar deve permitir esclarecer:

- as razões pelas quais foram escolhidas as estruturas de dados;
- a capacidade de modularidade que a solução encontrada apresenta;
- qual a complexidade (em termos de ocupação de memória) das representações escolhidas e a complexidade (em termos de tempo) das funções implementadas.

5 Cronograma

Sugere-se o desenvolvimento do projeto com base nas seguintes fases (ver figura 1):

Fase 1
1: Definição dos módulos da aplicação, incluindo a definição da interface de cada componente (i.e, definição dos ficheiros .h de cada módulo)
2: Implementação da funcionalidade relativa ao processamento dos ficheiros DBLP

Fase 2A
3: Implementação das estruturas de dados e funções básicas para gestão da informação (inserção, etc...)

Fase 2B
4: Implementação de requisitos complementares, incluindo otimização da aplicação e rede-social

Fases 1+2A+2B
5: Relatório final do projeto

A entrega far-se-á nas seguintes datas:

até 17.mar.2013: Aplicação, escrita em C, que efetua a leitura dos registos DBLP e produz um ficheiro com estatística básica (itens da Fase 1).

até 28.abr.2013: Aplicação final, escrita em C e produzindo os relatórios especificados, incluindo os requisitos complementares (itens das Fases 2A e 2B), e relatório final.

Devem aproveitar-se as aulas e os períodos de atendimento para esclarecimento de dúvidas.

6 Avaliação

Neste ano letivo, não haverá grupos neste 1º projeto, i.e., os trabalhos em C são feitos **individualmente**. Cada aluno deverá entregar o seu próprio projeto.
A avaliação do funcionamento da aplicação em C será feita de forma maioritariamente automática. Para tal, será fundamental obedecer de forma rigorosa aos formatos indicados, pois a avaliação é feita com base em exemplos de entrada e na comparação da saída gerada pela aplicação com os resultados expectáveis.
Todos alunos terão acesso a um conjunto de exemplos e respetivos resultados, que deverão ser usados para teste da aplicação. A avaliação final recorrerá a um conjunto de testes, que incluirá outros exemplos, para além daqueles inicialmente disponibilizados.
As aplicações serão ainda comparadas para efeitos de deteção automática de plágio, usando os mais modernos e sofisticados mecanismos à disposição. Os casos que configurem situações de plágio implicarão a reprovação de todos os alunos envolvidos (i.e., quer os que plagiaram, quer os que deixaram plagiar o seu trabalho).
Os pesos das componentes de avaliação são os seguintes:

1. requisitos nucleares — 13 valores

   (a) Fase 1: 5 valores
   (b) Fase 2A: 8 valores

2. requisitos complementares (Fase 2B) — 4 valores

   (a) Otimização: 2 valores
   (b) Rede social: 2 valores

3. relatório — 3 valores

Os requisitos nucleares são de implementação obrigatória e quem não os entregar até às respetivas datas limite fica imediatamente "reprovado". Uma avaliação inferior a 10 valores neste projeto equivale à reprovação na UC LI3. Nesses cenários, o estudante não será avaliado no 2º trabalho, mesmo que nele participe. Note-se que um aluno que opte por não abordar os requisitos complementares está limitado a uma nota máxima de 16 valores.

A Detalhes das entradas DBLP a suportar

As entradas bibliográficas de artigos publicados em revistas têm os seguintes campos:

1. número inteiro;
2. lista de autores separados por "," e terminada por ":";
3. título terminado por ".";
4. nome da revista, que inclui uma sigla indicada entre parêntesis;
5. volume da revista (número inteiro);
6. número da revista indicado entre parêntesis e terminado por ":";
7. página inicial e página final separadas pelo caracter "-";
8. ano, indicado entre parêntesis.

O seguinte exemplo mostra a concretização desse formato:

> 6 Hugo Paredes, Fernando Mário Martins: Social interaction regulation in virtual web environments using the Social Theatres model. J. Network and Computer Applications (JNCA) 35(1):3-19 (2012)

As entradas bibliográficas de artigos publicados em conferências têm os seguintes campos:

1. número inteiro;
2. lista de autores separados por "," e terminada por ":";
3. título terminado por ".";
4. nome da conferência, tipicamente a sigla;
5. ano terminado por ":";
6. página inicial e página final separadas pelo caracter "-".

Os exemplos seguintes mostram a concretização desse formato:

> 33 Diogo Telmo Neves, Tandy Warnow, João Luís Sobral, Keshav Pingali: Parallelizing SuperFine. SAC 2012:1361-1367

> 879 Kiyoshi Kiyokawa, Masahide Hatanaka, Kazufumi Hosoda, Masashi Okada, Hironori Shigeta, Yasunori Ishihara, Fukuhito Ooshita, Hirotsugu Kakugawa, Satoshi Kurihara, Koichi Moriyama: Owens Luis - A context-aware multi-modal smart office chair in an ambient environment. VR 2012:1-4

Repare-se que devem ser considerados vários tipos de caracteres (á, É, à, ë, û, õ, ç, ß, ø, æ, &, λ). Há que ter em atenção que alguns títulos têm caracteres de vários tipos, nomeadamente artigos de carácter mais matemático. Alguns títulos têm um ":", que não deve ser confundido com o caracter que indica o fim da lista de autores.
As entradas que não seguem qualquer destes dois formatos devem ser ignoradas, ou seja, deve ignorar-se todas as entradas que não correspondam a artigos. Devem, em particular, ser ignoradas as entradas sem autores. Devem também ser ignoradas as entradas que incluam no título os seguintes textos (com eventuais substituições de letras minúsculas por maiúsculas): "editorial", "preface", "errata", "obituary", "in memory of" e "isbn". Finalmente, devem ser ignoradas as entradas em que o número de páginas seja inferior ao valor indicado pelo utilizador em (C).
Os seguintes exemplos mostram entradas que devem ser ignoradas:

> 278 Luís Gomes, Victor Khomenko, João M. Fernandes: 10th International Conference on Application of Concurrency to System Design, ACSD 2010, Braga, Portugal, 21-25 June 2010 ACSD 2010 [não é artigo]
>
> 370 : Probabilistic properties of B-splines. Computer-Aided Design (CAD) 25(10):677 (1993) [não tem autores]
>
> 3423 István Györi, David W. Reynolds: Preface. Computers & Mathematics with Applications (CMA) 64(7):2159 (2012) [contém "Preface"]
>
> 860 Thierry Denoeux: Special issue in memory of Philippe Smets (1938-2005). Int. J. Approx. Reasoning (IJAR) 48(2):349-351 (2008) [contém "in memory of"]
>
> 1523 Irek Defée: Exploration of Visual Data by: Xiang Sean Zhou, Yong Rui, Thomas S. Huang; Kluwer Academic Publishers, 2003, 187 pp. ISBN 1-4020-7569-3. Signal Processing (SIGPRO) 84(2):441-442 (2004) [contém "ISBN"]

A gramática correspondente às entradas DBLP a suportar está especificada na seguinte figura.

Entrada de Artigo de Revista

<inteiro> <lista_autores> ":" <titulo> "." <nome_revista> "("
<sigla> ")" <volume_revista> "(" <numero_revista> ")" ":"
<pag_inicial> "-" <pag_final> "(" <ano> ")"

Entrada de Artigo de Conferência

<inteiro> <lista_autores> ":" <titulo> "." <nome_conf> <ano>
":" <pag_inicial> "-" <pag_final>

sendo:

<lista_autores> ::- string | string "," <lista_autores>

<inteiro> ::- string

<titulo> ::- string

<nome_revista> ::- string

<nome_conf> ::- string

<sigla> ::- string

<volume_revista> ::- <inteiro>

<numero_revista> ::- <inteiro>

<pag_inicial> ::- <inteiro>

<pag_final> ::- <inteiro>

<ano> ::- <inteiro>