

# INF1301 Programação Modular

Período: 2019-2

Prof. Flavio Bevilacqua

2o. Trabalho

Data de divulgação: 09 de setembro (segunda-feira)

Data de entrega: 07 de outubro (segunda-feira)

## 1. Descrição do trabalho do período

O objetivo do trabalho do período é desenvolver uma aplicação (“Labirinto”) que permita construir um labirinto e disponibilize um resolvidor para o mesmo. Será possível construir manualmente um labirinto através de funções específicas (com apenas quatro direções a serem tomadas a cada decisão: norte, sul, leste, oeste) e uma função adicional deverá imprimir o caminho da entrada até a saída. O grupo deverá dimensionar o projeto de acordo com o tempo disponível. As funções, forma de impressão, interfaces, módulos, arquitetura da aplicação, etc... ficarão sob responsabilidade do grupo. No Trab 2 serão projetados, implementados e testados separadamente alguns Módulos da aplicação. No Trabalho 3 haverá o projeto, implementação e teste dos módulos restantes, bem como o Principal e assim será concluída a aplicação. A função que “resolve” o Labirinto apresentando o caminho da entrada à saída (ou informa não haver solução) deverá ser implementada utilizando o algoritmo apresentado em aula.

## 2. Descrição do segundo trabalho

Para este segundo trabalho, o grupo deverá realizar as seguintes tarefas abaixo:

- Elabore uma documentação de requisitos completa para a aplicação “Labirinto”. Este servirá de insumo para o projeto da arquitetura.
- Elabore o **diagrama de arquitetura completo da aplicação “Labirinto”**. Este deve conter todos os módulos e todas as interfaces (inclusive com assinaturas completas de cada função de acesso). Lembre que a aplicação a ser completada no Trabalho 3 rodará fora do arcabouço com uma interface com o usuário, ou seja, terá um módulo principal. Será obrigatório que a arquitetura da aplicação tenha pelo menos **dois tipos abstratos de dados (um pode utilizar o outro) além de um módulo centralizador principal. É PROIBIDA a utilização de LISTAS ENCADEADAS e QUALQUER ESTRUTURA ESTÁTICA (por exemplo: VETORES ESTÁTICOS)**.
- Elabore o modelo físico completo (junto com as assertivas e um exemplo físico do labirinto) da estrutura que armazenará o labirinto completo na memória.
- Implemente e teste separadamente **dois** dos tipos abstratos de dados projetados. Cada módulo deverá ser acoplado ao arcabouço e ao final o trabalho 2 englobará dois executáveis, um para cada tipo abstrato de dados testado.

## 3. Entrega do Trabalho

O trabalho deve ser feito em grupos de dois ou três alunos. Os programas devem ser redigidos em “C”. Não será aceita nenhuma outra linguagem de programação. Todos os programas devem estar em conformidade com os padrões dos apêndices de 1 a 10 do livro-texto da disciplina. Em particular, os módulos e funções devem estar devidamente especificados.

Recomenda-se fortemente a leitura do exemplo e do texto explanatório do arcabouço. Além de mostrar como implementar um teste automatizado dirigido por script, ilustra também as características de um programa desenvolvido conforme os padrões do livro.

O trabalho deve ser enviado por e-mail em um único arquivo **.zip** (codificação do attachment: **MIME**). Veja os Critérios de Correção de Trabalhos contidos na página da disciplina para uma explicação de como entregar.

O arquivo **.zip** deverá conter:

- Os arquivos-fonte dos diversos módulos que compõem o programa.
- Os arquivos de script de teste completos desenvolvidos pelo grupo.
- Os arquivos batch (**.bat**) que coordenam a execução dos testes.
- programa executável (construto): **TRAB2-1.EXE (para o teste do primeiro TAD) e TRAB2-2.EXE (para o teste do segundo TAD)**
- Um arquivo **LEIAME.TXT** contendo a explicação de como utilizar o(s) programa(s).
- Um documento **.pdf** contendo o Diagrama de Arquitetura da Aplicação
- Um documento **.pdf** contendo a documentação de Requisitos da Aplicação
- Um documento **.pdf** contendo o Modelo Físico Estrutural do Labirinto, bem como Assertivas e Exemplo Físico.
- Tantos arquivos **RELATORIO-nome.TXT** quantos forem os membros do grupo. O tema **nome** deve identificar o membro do grupo ao qual se refere o relatório. Estes arquivos devem conter uma tabela de registro de trabalho organizada como a seguir:

**Data ; Horas Trabalhadas ; Tipo Tarefa ; Descrição da Tarefa Realizada**

Na descrição da tarefa redija uma explicação breve sobre o que o componente do grupo fez. Esta descrição deve estar de acordo com o Tipo Tarefa. Cada Tipo Tarefa identifica uma natureza de atividade que deverá ser discriminada explicitamente, mesmo que, durante uma mesma sessão de trabalho tenham sido realizadas diversas tarefas.

Os tipos de tarefa são:

- ◆ estudar
- ◆ especificar os módulos
- ◆ especificar as funções
- ◆ revisar especificações
- ◆ projetar
- ◆ revisar projetos
- ◆ codificar módulo
- ◆ revisar código do módulo
- ◆ redigir script de teste
- ◆ revisar script de teste
- ◆ realizar os testes
- ◆ diagnosticar e corrigir os problemas encontrados

**OBS IMPORTANTE: NÃO É OBJETIVO DO TRABALHO 2 COMPLETAR A APLICAÇÃO DO LABIRINTO NEM IMPLEMENTAR O MÓDULO PRINCIPAL. NADA SERÁ INTEGRADO NESTE MOMENTO.**

**Observações:**

- Dica: Preencha esta tabela de atividades ao longo do processo. **NÃO DEIXE PARA ÚLTIMA HORA, POIS VOCÊ NÃO SE LEMBRARÁ DO QUE FEZ TAL DIA, TAL HORA.** Com relatórios similares a esse você aprende a planejar o seu trabalho.
- **Importante:** O arquivo **ZIP**, DEVERÁ CONTER SOMENTE OS ARQUIVOS RELACIONADOS A ESTE TRABALHO. Caso o arquivo enviado contenha outros arquivos além dos acima enumerados (por exemplo: toda pasta do arcabouço, arquivos **.bak**, arquivos de trabalho criados pelo ambiente de desenvolvimento usado, etc.) o grupo perderá 2 pontos. Gaste um pouco de tempo criando um diretório de distribuição e um **.bat** que copia do diretório de desenvolvimento para este diretório de distribuição somente os arquivos que interessam. Verifique se esta cópia está realmente completa! A mensagem de encaminhamento deve ter o assunto (subject) **INF1301-Trab02-idGrupo** conforme o caso. O tema **idGrupo** deve ser formado pelas iniciais dos nomes dos membros do grupo. O texto da mensagem deve conter somente a lista de alunos que compõem o grupo (formato: número de matrícula, nome e endereço do e-mail). Perde-se 2 pontos caso não seja encaminhado desta forma. Mais detalhes podem ser encontrados no documento Critérios de Correção dos Trabalhos disponível na página da disciplina.
- O programa será testado utilizando o programa compilado fornecido. Deve rodar sem requerer bibliotecas ou programas complementares. Assegure-se que a versão do programa entregue é uma versão de produção, ou seja, sem dados e controles requeridos pelo debugger (versão release).

#### 4. Critérios de correção básicos

Leia atentamente o documento Critérios de Correção dos Trabalhos enviado pelo professor. Muitas das causas para a perda substancial de pontos decorrem meramente da falta de cuidado ao entregar o trabalho.

**Não deixem para a última hora.  
Este trabalho dá trabalho!**