**INF2102 Projeto Final de Programação**

**Período: 2020.1**

**Coordenador: Prof. Marcos Kalinowski Informações gerais**

# Objetivo da disciplina

É objetivo do projeto final de programação averiguar se o aluno sabe empregar técnicas eficazes para especificar, projetar, desenvolver, controlar a qualidade e documentar programas, que tenham um nível de complexidade pelo menos mediano. Os programas devem ser confiáveis, úteis e utilizáveis e devem possuir elevada qualidade de engenharia. Na medida do possível, o trabalho final de programação deve estar relacionado com a dissertação de mestrado ou tese de doutorado do aluno.

# Natureza do trabalho

O **resultado** do projeto de programação pode ser *qualquer programa* que sirva para demonstrar a capacidade do aluno de *especificar*, *desenvolver e testar racionalmente* programas satisfazendo os objetivos acima.

O programa será **útil** na medida em que realize um serviço de interesse a alguém que não seja exclusivamente o aluno. A utilidade do programa será avaliada pelo *professor orientador* (ver mais adiante). Entende-se que o trabalho tem alguma relação com a dissertação de mestrado ou tese de doutorado a ser elaborada. São exemplos de aplicações:

* aplicações para engenharia
* componentes de compiladores
* ferramentas de apoio ao desenvolvimento de software
* frameworks
* interfaces gráficas animadas
* jogos eletrônicos
* produtos de software
* programas re-engenheirados
* sistemas “e-qualquer coisa”
* sistemas de supervisão e controle
* sistemas web
* sistemas big data
* sistemas vestíveis
* sistemas IoT

A **complexidade** envolve *tamanho* e *natureza* do problema resolvido pelo programa. Cabe ao professor orientador (ver mais adiante) assegurar a satisfação deste requisito.

A **especificação** deve deixar claro qual é a finalidade do programa. A tecnologia usada ao especificar pode ser qualquer. A especificação mínima que deve estar escrita antes de se redigir o programa deve responder as perguntas a seguir. Qual é a finalidade do programa? Quais são as interações entre os usuários (pessoas, sensores, atuadores, outros programas ou serviços) e o programa? A maneira mais simples (e fraca) para responder essa pergunta seria empregar, por exemplo, *user stories*. Caso o programa se relacione com a tese ou dissertação, qual seria o título tentativo dessas? Como será feita a aceitação do programa ao final do desenvolvimento?

A **garantia da qualidade** depende da especificação, arquitetura, projeto, programação e teste empregados. A presença e forma de redigir esses artefatos dependem do paradigma usado pela solução (p.ex. programação lógica, procedural, OO, dirigida por tabela, etc.). A documentação entregue ao final do projeto deve conter os artefatos listados adaptados ao paradigma usado. Deve também deixar claro que foi empregado um *controle da qualidade* (teste, inspeção) sistemático. O ideal é ter sido utilizada alguma forma de teste automatizado (ex. JUnit, CPPUnit, LuaUnit, Python unit testing framework, ou similares).

A **linguagem de programação** utilizada pode ser qualquer uma (ex. Lua, Lisp, Modula, C, C++, Java, Prolog, Python, Ruby, etc.). Também pode ser qualquer uma a plataforma utilizada (ex. Windows, Mac, Unix, iOs, celulares, tablets, etc.) ou o framework usado. Sendo de interesse do aluno, o trabalho pode ser desenvolvido utilizando alguma ferramenta CASE.

# Como iniciar

Antes de mais nada, encontre um *professor* *orientador* do projeto de programação e informe na SECRETARIA. Este **é necessariamente** um professor do Departamento de Informática da PUC-Rio. Não deve ser o coordenador da disciplina. Sugere-se que seja o orientador da tese ou dissertação.

Junto com o professor orientador defina o objetivo do trabalho. O ideal seria que isto fosse feito antes do início do semestre e que se relacione com a dissertação ou tese.

Envie ao professor coordenador da disciplina, prof. Marcos Kalinowski, até o dia **17/abril/2019**, um email (kalinowski@inf.puc-rio.br), contendo:

* o número de matrícula e o nome do aluno
* o nome do professor orientador
* o nome do projeto
* um resumo curto (um parágrafo) descrevendo o objetivo do projeto

O título da mensagem deve ser INF2102-Proposta-***NomeAluno***

* obviamente a parte ***NomeAluno*** deve ser substituída pelo respectivo nome.

# Como terminar

Entregar na SECRETARIA, até o dia **3/julho/2019**:

1. uma folha contendo (ver apêndice):
   * o número de matrícula e o nome do aluno
   * o nome do professor orientador
   * a **avaliação do trabalho** feita pelo professor orientador (tipicamente um parágrafo de texto). Esta avaliação deve dizer se:
   * o programa satisfaz os requisitos de complexidade assumidos para projetos finais de programação
   * o programa é confiável e utilizável
   * o programa atende aos objetivos funcionais traçados
   * se os testes foram abrangentes e satisfatórios
   * eventuais comentários a respeito do trabalho
   * assinatura do professor orientador

1. entregar na secretaria a documentação final do projeto. Recomenda-se a entrega de tudo em um **pendrive**, ***exceto a folha de avaliação que deve ser impressa e assinada pelo orientador***. Os arquivos contidos no **pendrive** ***não*** devem estar compactados. Não se esqueça de identificar o autor no **pendrive**. Tampouco se esqueça de anexar a folha de avaliação preenchida pelo orientador. A documentação deve conter:
   * + especificação do programa
     + objetivos, requisitos
     + especificação, por exemplo *use-cases*.
     + projeto do programa
     + arquitetura
     + critérios de projeto utilizados
     + diagramas de arquitetura e/ou segmentação do programa, por exemplo UML.
     + organização do programa (componentes, módulos, classes,...), por exemplo diagramas de classe UML.
     + diagramas de organização dos dados, por exemplo diagramas de modelagem de dados, ou entidade e relacionamentos.
     + código fonte **cuidadosamente comentado**
     + comentário inicial de cada módulo, identificando o autor
     + comentários cabeçalho de módulos, classes e funções
     + assertivas para dados e procedimentos, procure utilizar *design by contract*
     + pseudo instruções
     + procure estabelecer e/ou adotar padrões de programação.
     + roteiro de teste efetuado, composto de:  critérios de teste utilizados
     + descrição dos casos de teste
     + na medida do possível procure utilizar testes automatizados, neste caso adicione os scripts de teste e os logs gerados pelo teste automatizado  documentação para o usuário

O formato da documentação técnica varia em função do domínio do problema e da linguagem de programação utilizada. Não obstante, o programa deverá ser especificado e projetado antes de ser implementado. Ou seja, a especificação, a organização e a composição do programa deverão estar definidas antes de iniciar a codificação. Antes de iniciar deve também ser planejado e descrito como será realizado o controle da qualidade do programa, usualmente como será testado. O plano deve ser capaz de atestar que o programa é confiável e atinge os objetivos especificados. Evidentemente o controle da qualidade, usualmente realizado por testes, depende da natureza do programa, da linguagem e do ambiente de execução. **Critério de avaliação**

**Não serão aceitos** trabalhos sem a folha de avaliação produzida pelo professor orientador.

As notas máximas serão:

10 trabalho satisfatório e entregue no prazo

8 trabalho satisfatório e entregue até **31/agosto/2020**. Esta data está relacionada com a data limite estabelecida pela PUC-Rio para preenchimento de graus IN do semestre anterior. Já prevê uma folga para a correção dos inúmeros trabalhos.

zero qualquer outra situação. *Cabe salientar que, segundo o regulamento da pósgraduação, se um zero nesta disciplina fizer com que a média das notas* ***do semestre*** *caia abaixo de seis, o aluno será automaticamente desligado do programa de pósgraduação.*

**Se você acha que não vai conseguir terminar no prazo**, cancele a disciplina. Não se matricule em *Projeto Final de Programação* sem ter orientador e saber qual será o trabalho!

Trabalhos que não forem entregues até **3/julho/2020**, recebem **automaticamente** IN (incompleto). Trabalhos que não forem entregues até **31/agosto/2020** recebem automaticamente zero.

Causam a perda de pontos:

* entrega atrasada. Para evitar a perda de pontos só se matricule depois que souber quem é o orientador, qual é o trabalho, e se você tem condições de terminar o trabalho no semestre. Afinal saber planejar faz parte da vida profissional. Se durante o semestre você perceber que não conseguirá terminar, cancele a disciplina. Não se esqueça que o calendário da PUC-Rio determina um prazo para o cancelamento de disciplinas.
* inexistência de avaliação feita pelo orientador, **o trabalho não será aceito sem a avaliação do professor**.
* a avaliação feita pelo orientador, caso a nota sugerida não seja dez.
* inexistência de especificação.
* inexistência de projeto.
* inexistência de um plano de controle da qualidade. Caso esse controle seja realizado usando testes, deve estar explicitado o método de teste utilizado, bem como a suíte de teste utilizada. Dê preferência a testes automatizados.
* inexistência de evidência de que foi realizado o controle da qualidade segundo o plano. Procure produzir um laudo de teste. O teste automatizado deve produzir arquivos log.
* inexistência de documentação para o usuário.
* código mal organizado e/ou mal comentado.

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO**

**DEPARTAMENTO DE INFORMÁTICA**

**INF 2102 - PROJETO FINAL DE PROGRAMAÇÃO**

**FOLHA DE AVALIAÇÃO**

**NOME DO ALUNO(A):** **Nº MATRÍCULA:**

**NOME DO ORIENTADOR:**

**TÍTULO DO PROJETO:**

**PALAVRAS-CHAVE:**

**AVALIAÇÃO DO TRABALHO**

|  |  |  |
| --- | --- | --- |
| Sim Não |  | Este trabalho satisfaz os requisitos de complexidade assumidos para projetos finais de programação |
| Sim Não |  | O programa é confiável e utilizável |
| Sim Não |  | O programa atende aos objetivos funcionais traçados |
| Sim Não |  | Os testes foram abrangentes e satisfatórios |

**OBSERVAÇÕES**

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |

|  |
| --- |
|  |

**NOTA SUGERIDA:**

Rio de Janeiro,

Professor Orientador