



Professor Celso Olivete Júnior

- Bacharelado em Ciência da Computação (Unoeste-2002)
- Mestrado e Doutorado em Engenharia Elétrica - Área: Visão Computacional (USP-SC-2005/2009)
- Áreas de interesse e atuação:
 - visão computacional
 - desenvolvimento Web
 - compiladores



Site do Curso

www.fct.unesp.br/docentes/dmec/olivete/tc

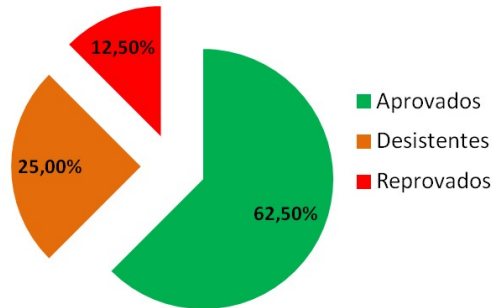
- slides, exercícios, notas e demais materiais estarão disponíveis no site

- Envio de trabalhos e dúvidas através do email

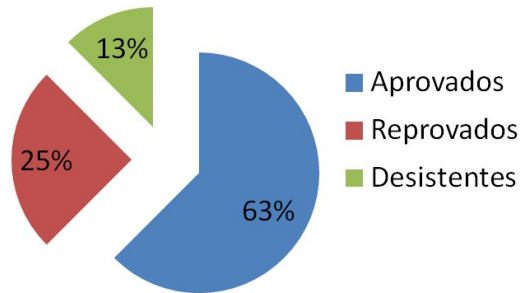
olivete@fct.unesp.br



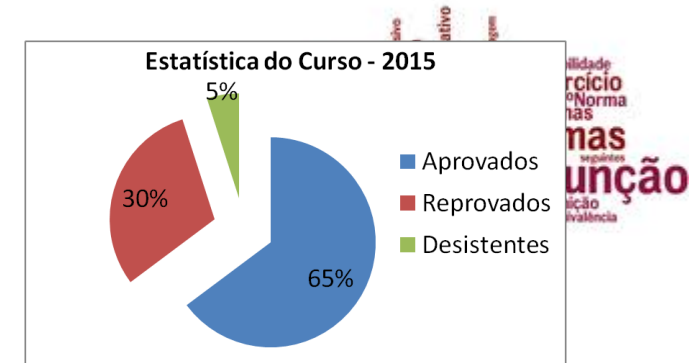
Estadística do Curso - 2013



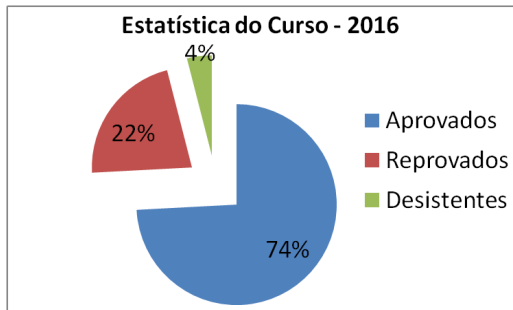
Estadística do Curso - 2014



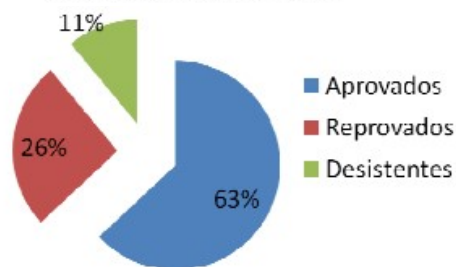
Estadística do Curso - 2015



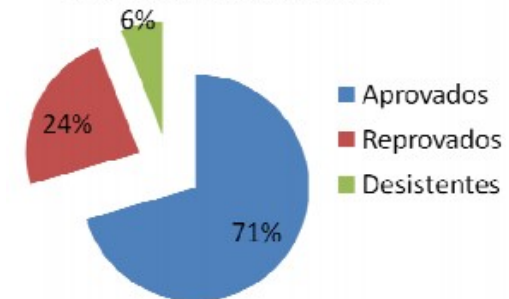
Estadística do Curso - 2016



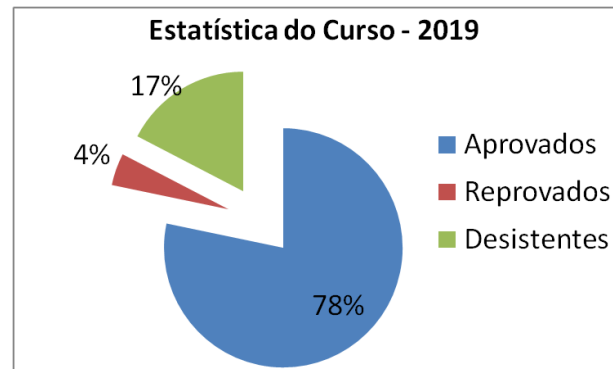
Estadística do Curso - 2017



Estadística do Curso - 2018



Estadística do Curso - 2019

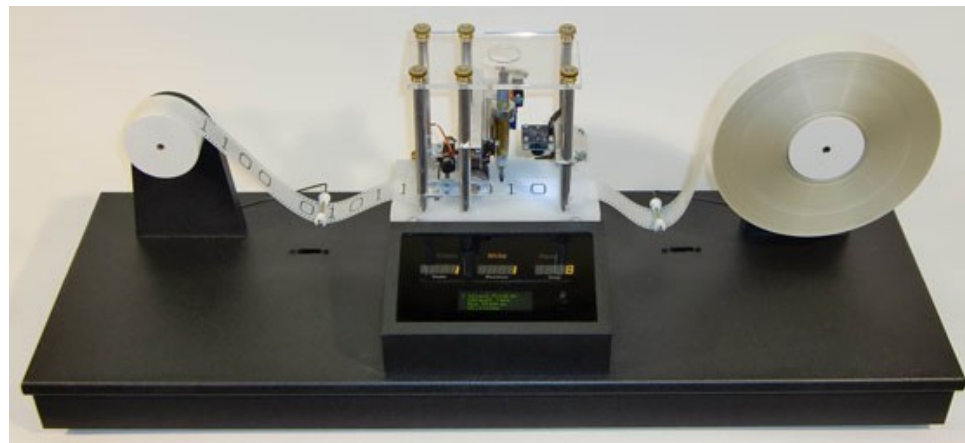


Celso Olivete Júnior

A disciplina Teoria da Computação



- **Máquina de Turing:** modelo construído com circuitos digitais

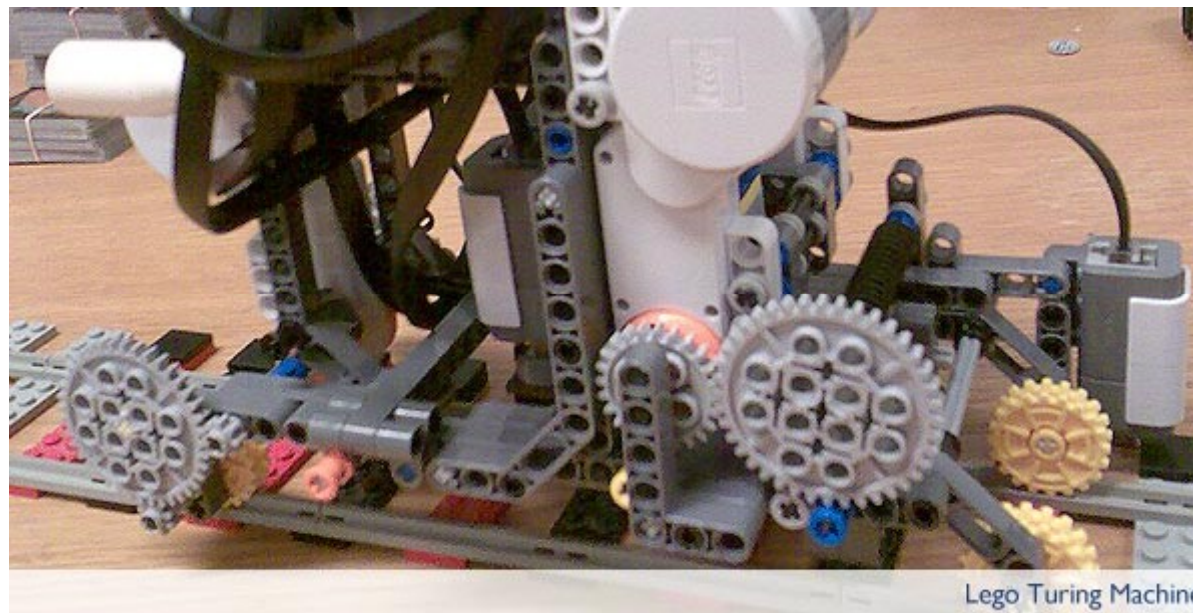


<http://www.youtube.com/watch?v=E3keLeMwfHY>

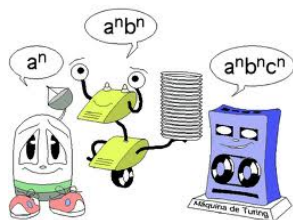


A disciplina Teoria da Computação

- **Máquina de Turing:** modelo construído com lego



<http://www.youtube.com/watch?v=-getuN11LbI>



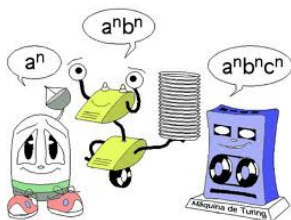
A disciplina Teoria da Computação



- **Máquina de Turing:** reportagem...



<http://www.youtube.com/watch?v=yIluxaHLOv0>





Programa detalhado

1. Introdução e conceitos básicos:

- Símbolos, cadeias, conjuntos e suas operações;

2. Programas, Máquinas e Computação:

- Tipos de Programas, Máquinas, Computações e Funções Computadas, Equivalência de Programas e Máquinas, Verificação da Equivalência Forte de Programas

3. Máquinas Universais:

- Codificação de Conjuntos Estruturados, Máquina de Registradores - Norma, Máquina Norma como Máquina Universal, Máquina de Turing, Outros Modelos de Máquinas Universais, Modificações sobre as Máquinas Universais, Hierarquia de Classes de Máquinas e Hipótese de Church

4. Computabilidade

- Classes de Solucionabilidade de Problemas



Avaliação

- Avaliação 1: 23-24/04 Avaliação 2: 25-26/06 EXAME: 02/07
- As notas de todas as atividades - entre 0 (zero) e 10,0 (dez) - serão atribuídas individualmente, mesmo em atividades em grupo;
- A média final será calculada da seguinte maneira:
 $MA = (NP1 + NP2)/2$
 $Mt = (NT1 + NT2 + \dots + NTn) / n$
 $MT = (7 * NPJ + 3 * Mt)$
- Média Final:
 $MF = (7 * MA + 3 * MT) / 10$ SE E SOMENTE SE $(MA \geq 5$ E $MT \geq 5)$
- Caso contrário ($MA < 5$ OU $MT < 5$)
 $MF =$ Menor Nota (MA ou MT)
- Sendo:
MF = Média Final.
MA = Média de Provas
MP = Média de Trabalhos e Projeto
Mt = Média de Trabalho (Listas de Exercícios)
NPJ = Nota Projeto
MT = Média final dos trabalhos (parte prática).

• Atendendo a RESOLUÇÃO UNESP 75/2016, que extingue o Regime de Recuperação e implanta o Processo de Recuperação, composto por: ações pedagógicas, no qual serão propostas atividades extra sala, durante o semestre letivo objetivando minimizar as dificuldades de aprendizagem dos estudantes identificados com baixo rendimento; e a Realização do Exame Final, constituído por uma avaliação contendo todo o conteúdo programático, teórico e das atividades práticas. Todos os alunos com Média Semestral (MS) menor do que 5.0 (cinco) poderão fazer o Exame Final. Desta forma, a nova Média Final do aluno será obtida pela média aritmética simples entre a Média Semestral e a nota do Exame Final, que deverá ser igual ou maior que 5.0 (cinco) para aprovação:

$$\text{Média Final} = (\text{Média Semestral} + \text{Exame Final}) / 2$$

se Média Final \geq 5: "Aprovado"; caso contrário: "Reprovado"



Avaliação

- A “cola” ou plágio em provas, exercícios ou atividades práticas implicará na atribuição de nota zero para todos os envolvidos. Dependendo da gravidade do incidente, o caso será levado ao conhecimento da Coordenação e do Conselho do Departamento, para as providências cabíveis. Na dúvida do que é considerado cópia ou plágio, o aluno deve consultar o professor antes de entregar um trabalho.



BIBLIOGRAFIA COMPLEMENTAR

- KFOURY, A. F., *A programming approach to computability*, Springer.
- MINSKY, M., *Computation: finite and infinite machines*, Prentice-Hall
- GERSTING, J. L., *Fundamentos Matemáticos para a Ciência da Computação*. 4^a ed., Editora LTC, 2001.
- PIPPENGER, N., *Theories of Computability*. Cambridge Univ. Press, 1997.
- KFOURY, A. F., *A programming approach to computability*, Editora Springer-Verlag.

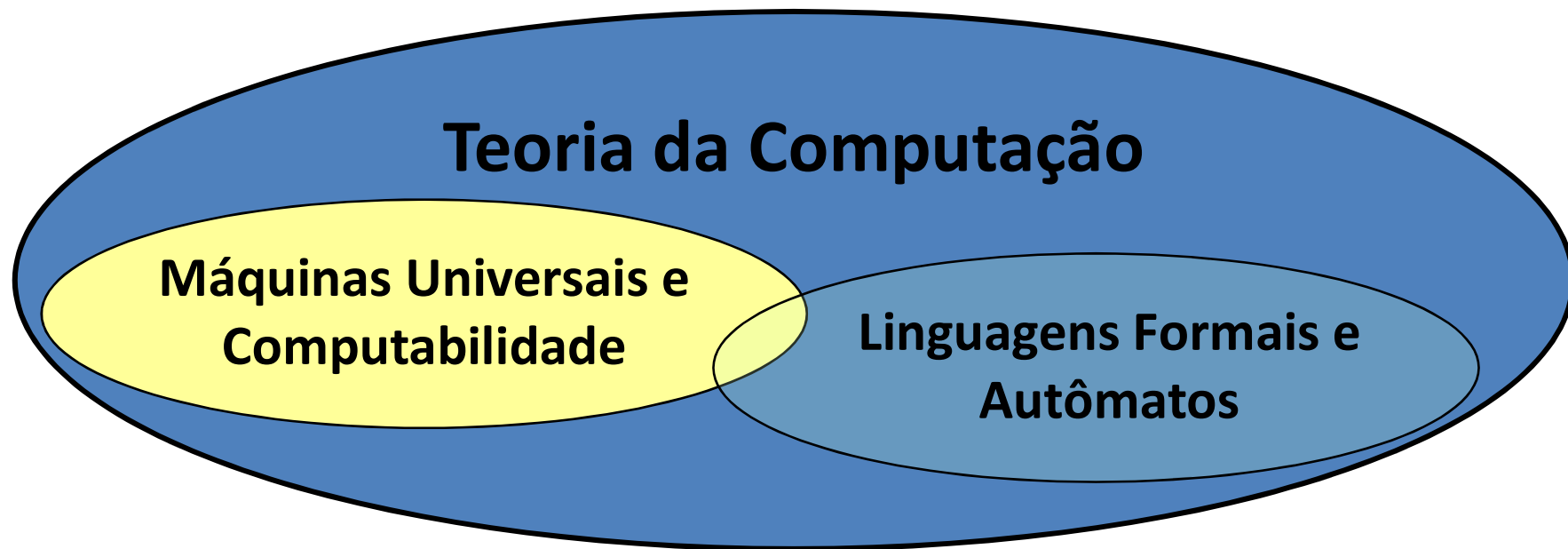
BIBLIOGRAFIA BÁSICA

- DIVERIO, T. A.; MENEZES, P. B. *Teoria da Computação: Máquinas Universais e Computabilidade*. Porto Alegre: Sagra-Luzzatto, 2000.
- HOPCROFT, J. E.; ULLMAN, J. D.; MOTWANI, R. *Introdução à Teoria de Autômatos, Linguagens e Computação*. Rio de Janeiro: Elsevier, 2002.



Visão geral da disciplina

- Dentro da Teoria da Computação encontra-se duas linhas de estudo:





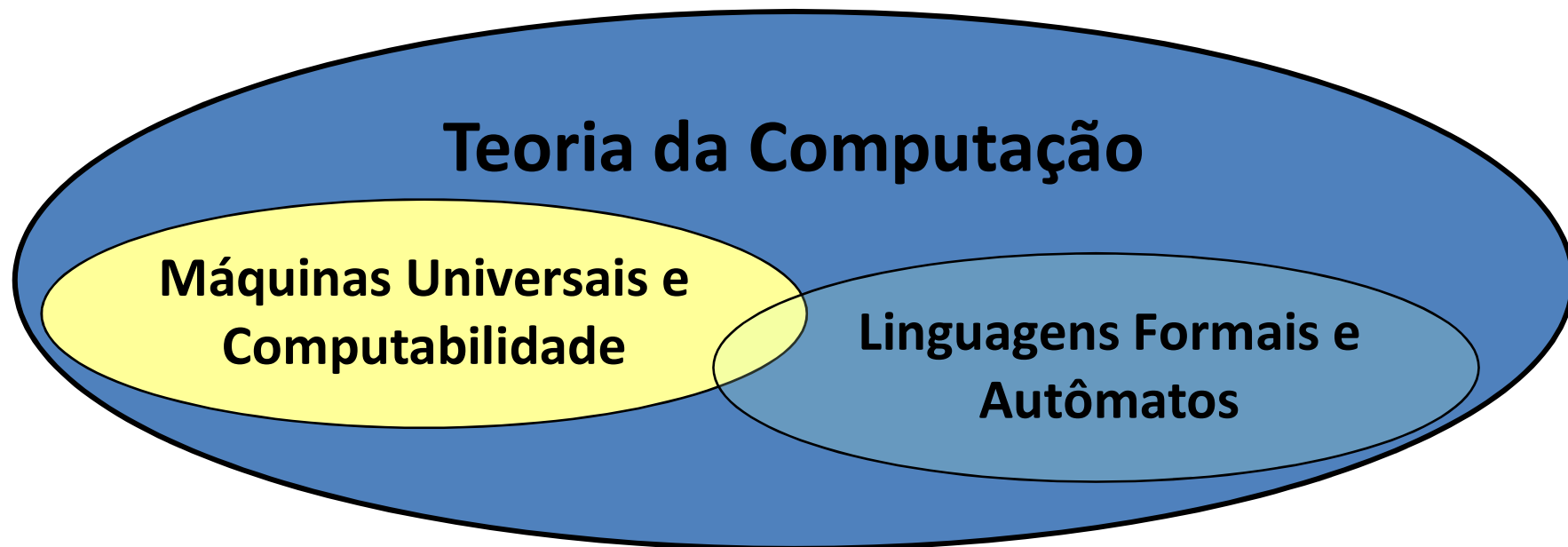
Visão geral da disciplina

- LFA
 - trata das definições e propriedades de modelos matemáticos que tem um papel fundamental em várias áreas da Computação como o processamento de textos, compiladores, definição de LP's, dentre outras.



Visão geral da disciplina

- Dentro da Teoria da Computação encontra-se duas linhas de estudo:





Visão geral da disciplina

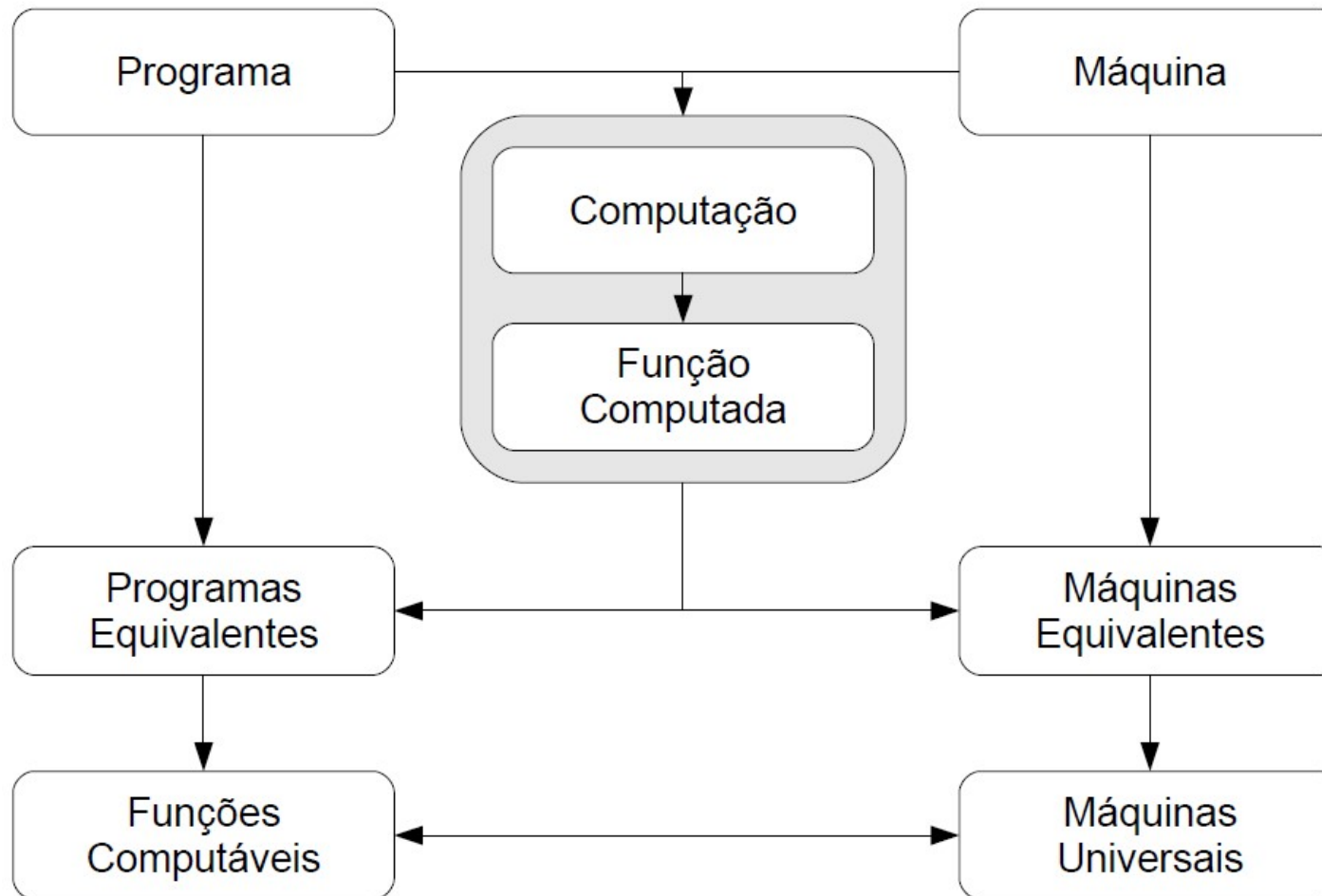
- Máquinas universais:

- Se um problema algorítmico não pode ser solucionado por uma máquina universal, então não existe uma solução computável para ele.

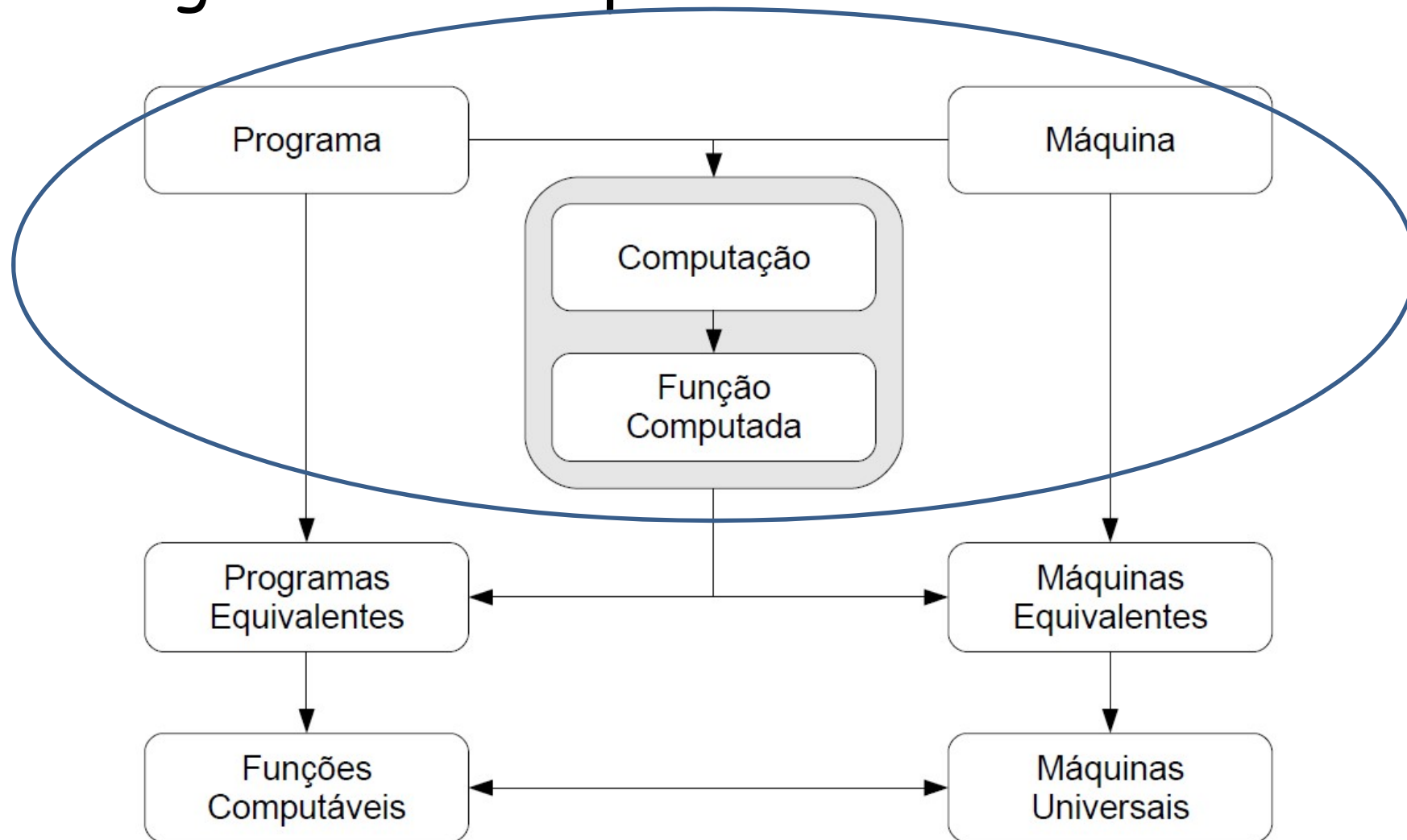
- Computabilidade

- classifica os problemas em solúveis, parcialmente solúveis e insolúveis, e se forem problemas de decisão, em problemas decidíveis, parcialmente decidíveis e indecidíveis.

Visão geral da disciplina



Visão geral da disciplina





Programa, máquina, computação e função computada

- **Programa:** *"Conjunto estruturado de instruções que capacitam uma máquina a aplicar sucessivamente certas operações básicas e testes sobre dados iniciais fornecidos, com o objetivo de transformar estes dados numa forma desejável."*



Programa, máquina, computação e função computada

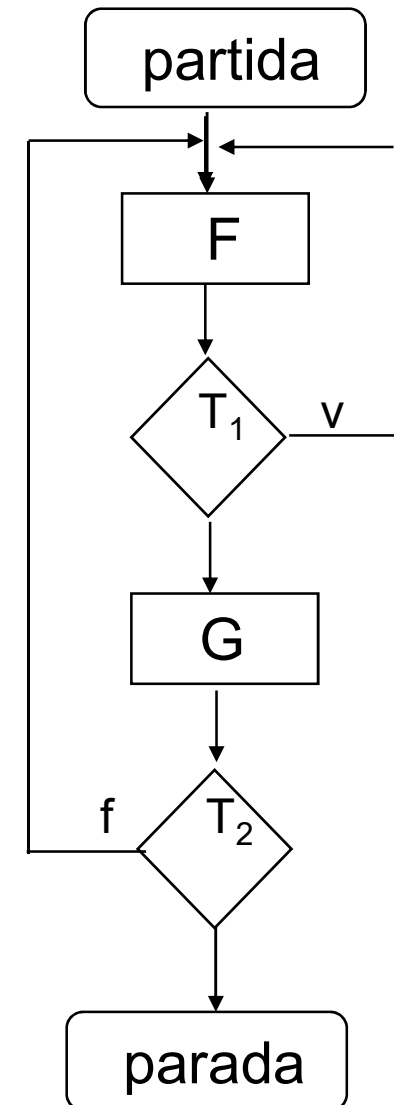
- um programa deve possuir uma estrutura de controle de operações e testes.
 - Estruturação Monolítica
 - Estruturação Iterativa
 - Estruturação Recursiva

Programa, máquina, computação e função computada

- Estruturação Monolítica - exemplo

Não faz uso explícito de mecanismos como iteração, subdivisão ou recursão.

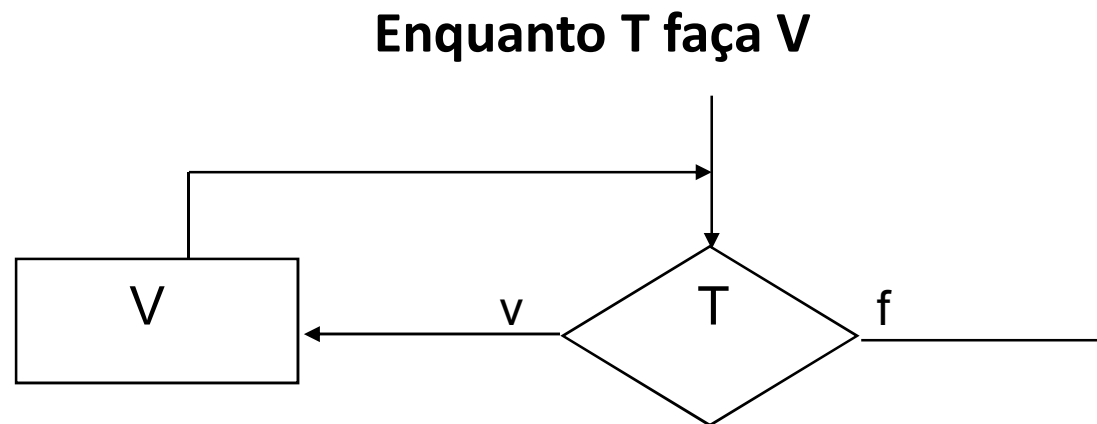
- 1: faça F vá_para 2
- 2: se T_1 então vá_para 1 senão vá_para 3
- 3: faça G vá_para 4
- 4: se T_2 então 5 (✓) senão vá_para 1



Programa, máquina, computação e função computada

- Estruturação Iterativa - exemplo

Substitui desvios incondicionais por estruturas de testes ou repetições resultando em uma melhor estruturação de desvios





Programa, máquina, computação e função computada

- Estruturação Recursiva - exemplo

Admite a definição de sub-rotinas recursivas

P é R; S onde

R def F; (se T então R senão G; S),

S def (se T então ✓ senão F; R)



Programa, máquina, computação e função computada

- Máquina

- Cabe a máquina dar significado aos identificadores das operações e testes
- exemplos: Norma, Post, Turing



Programa, máquina, computação e função computada

- **Computação**

- É um histórico do funcionamento da máquina para o programa, considerando um valor inicial.

- **Função computada**

- É o resultado obtido após o término da computação (finita)



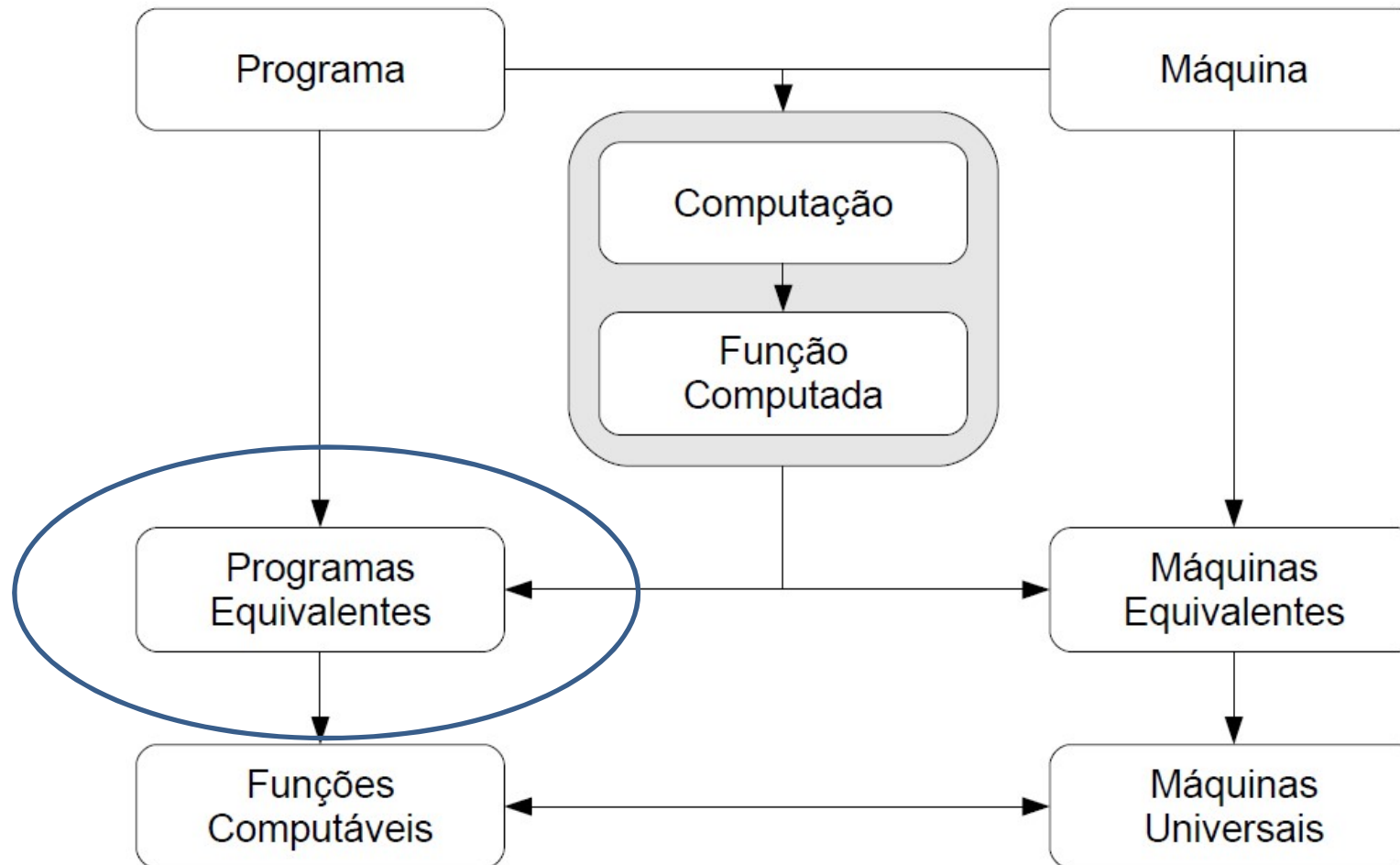
Programa, máquina, computação e função computada

- usando noções de programa, máquina e computação é estudado o conceito de computabilidade

- um programa para uma máquina pode induzir a uma computação

- se a computação for finita, então é definida a função computada (resultado) por esse programa nessa máquina
→ descreve o que o programa faz.

Visão geral da disciplina



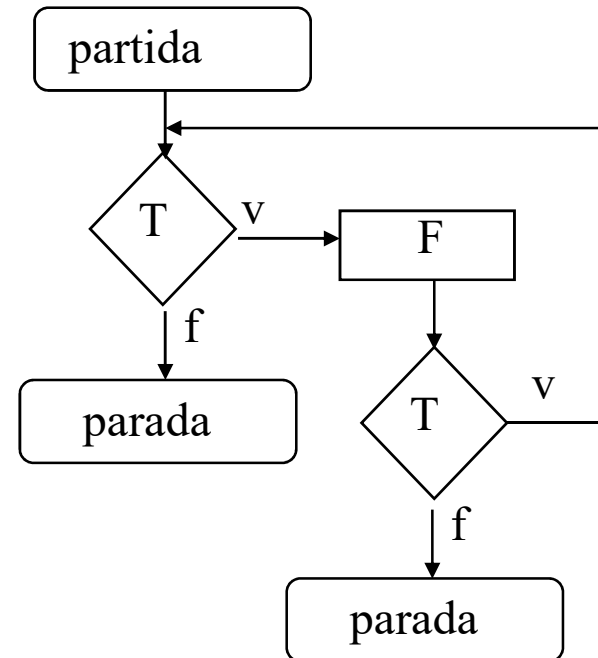
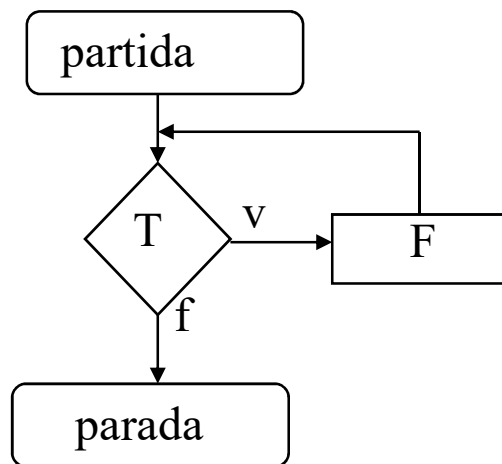


Equivalência de programas

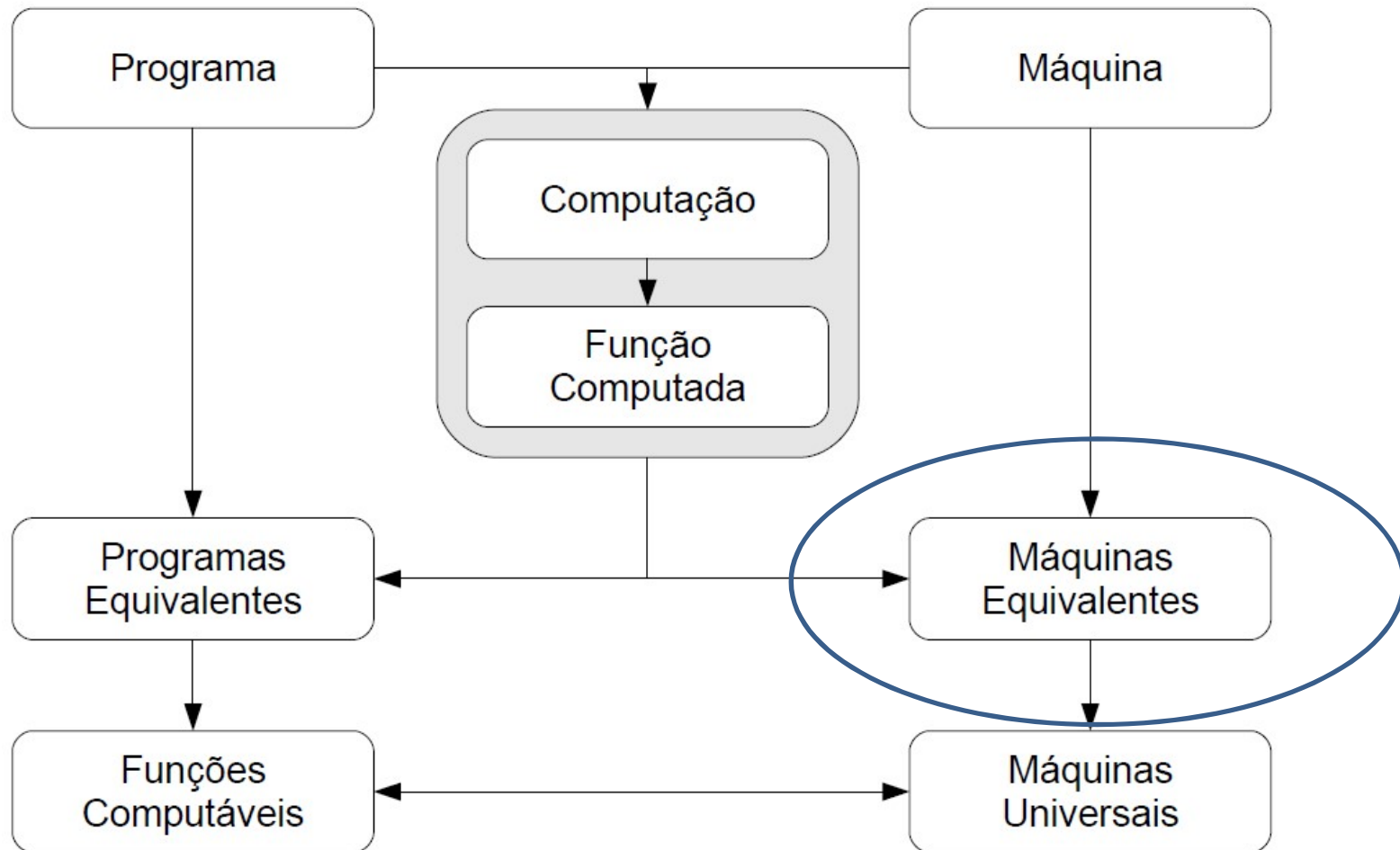
- Se dois programas, em uma máquina, possuem a mesma função computada (resultado), ou seja, computam a mesma função, então eles são **equivalentes**.
- O conceito de equivalência é utilizado para **otimizar o programa** → eliminar instruções desnecessárias

Equivalência de programas

- Exemplo



Visão geral da disciplina

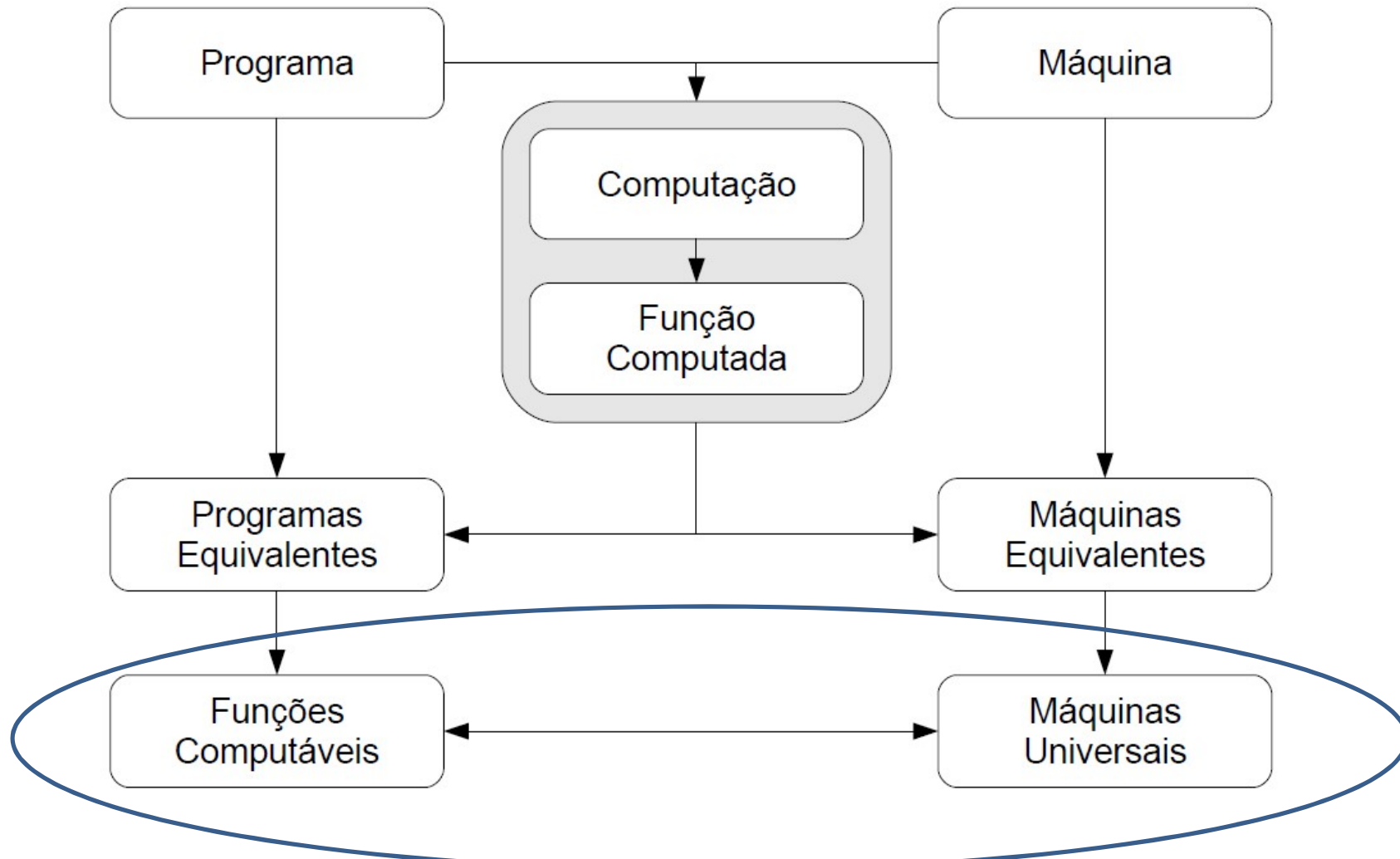




Equivalência de máquinas

- Se duas máquinas simulam-se mutuamente, é porque elas são equivalentes. Nesse caso, ambas tem o mesmo poder computacional.

Visão geral da disciplina





Funções computáveis e máquinas universais

- uma máquina é universal se toda função computável puder ser executada nela
 - uma função computável é aquela que pode ser processada numa Máquina de Turing ou equivalente (Post, Norma, Pilhas)



Visão geral da disciplina

- Teoria da Computação provê conceitos e princípios que ajudam a entender a natureza geral da computação
 - utiliza-se de computadores abstratos para resolução de problemas
- Para modelar o hardware do computador pode-se utilizar o conceito de *AF's*, que possuem as características de um computador digital - entrada, tomada de decisão, armazenamento e saída.



Visão geral da disciplina

- após estudar os conceitos de LFA torna-se necessário um estudo aprofundado sobre questões pertinentes a TC:
 - quais são as capacidades e limitações fundamentais dos computadores?
 - o que pode e o que não pode ser resolvido pelos computadores?
 - o que faz alguns problemas serem mais computacionalmente difíceis do que os outros?
- ao tentar responder essas questões, exploram-se os conceitos de **computabilidade**, **decidibilidade**, **reduzibilidade** e complexidade.

Visão geral da disciplina

- **Computabilidade, decidibilidade, redutibilidade** → conceitos que serão estudados utilizando modelos matemáticos, capazes de representar o que é uma computação

