

PLANO DE AULAS
ANO LETIVO: 2018 – 1º SEMESTRE

PROFESSOR: Bruno Oliveira Silvestre

DADOS DA DISCIPLINA:

Código	Nome da disciplina	Carga Horária	
		Teórica	Prática
INF0334	Software Básico	0	64

EMENTA:

Visão geral de um sistema computacional. Linguagem de montagem. Mapeamento de estruturas de linguagens de alto nível para a linguagem Assembly. Pilha de execução e modelo de chamadas de função. Interrupções e seus tipos. “Link-edição” e bibliotecas.

OBJETIVOS GERAIS:

Dar ao aluno um conhecimento básico de linguagem de montagem e o conhecimento de como elementos de linguagens de mais alto-nível são mapeadas para linguagem de montagem.

RELAÇÃO COM OUTRAS DISCIPLINAS:

O aluno deverá aplicar os conhecimentos adquiridos nas seguintes disciplinas: Introdução à Programação e Arquitetura de Computadores.

CONTEÚDO PROGRAMÁTICO:

1. Revisão de um sistema computacional:
 - 1.1. Introdução - Hierarquia de abstrações em um computador: Hardware, Linguagem de Máquina, Sistema Operacional, Linguagens de Alto Nível, Aplicativos.
 - 1.2. Arquitetura Clássica: CPU, memória principal, memória secundária, dispositivos de Entrada/Saída.
 - 1.2.1. CPU: ciclo de execução. Registradores (PC, Flags, SP).
 - 1.2.2. Memória principal: bits e bytes; palavras e endereçamento.
 - 1.3. Armazenamento e Representação de tipos básicos:
 - 1.3.1. Base binária, base hexadecimal, inteiros, complemento a dois.

1.3.2.	Operações lógicas a nível de bits.
1.3.3.	Caracteres: código ASCII e UTF.
1.3.4.	Ponto flutuante: o padrão IEEE.
1.4.	Implementação de arrays e registros; "padding"; arrays e registros aninhados; cálculo de tamanho e de endereçamento.
2.	Mapeamento de estruturas de uma linguagem de alto nível (linguagem C) em instruções da linguagem de montagem (linguagem Assembly):
2.1.	Conceitos básicos: Labels, mnemônicos, variáveis (estáticas), strings.
2.2.	Transferência de dados: modos de endereçamento, alinhamento.
2.3.	Atribuição.
2.4.	Expressões aritméticas.
2.4.1.	Inteiro e Ponto flutuante.
2.5.	Expressões lógicas.
2.6.	Loops e condicionais.
3.	Pilha de execução e chamadas:
3.1.	Registro de ativação, suporte a chamadas recursivas.
3.2.	Modelo de chamada de função e passagem de parâmetros.
3.3.	Criação e eliminação de registros de ativação
4.	Interrupções; tipos de interrupções: instrução explícita, erros de execução (traps), e geradas pelo hardware (assíncronas).
5.	"Link-edição": o problema da (re)locação de um programa. Compilação independente; programa objeto e tabelas de relocação. Funcionamento de um ligador. Formatos de executável, bibliotecas dinâmicas e estáticas.
6.	Depuração básica de código.

PROCEDIMENTOS DIDÁTICOS:

Legenda	Descrição	Objetivo
AEX	Aula teórica	Transmitir conhecimento utilizando quadro ou slides.
PRA	Aula prática	Executar na prática o conhecimento teórico adquirido.
TG	Trabalho em grupo	Desenvolver a capacidade de comunicação oral e escrita. Capacidade de trabalhar em grupo.
ED	Estudo Dirigido	Desenvolver a capacidade analítica, capacidade de síntese, de avaliação crítica e de análise.
SE	Seminários	Desenvolver o raciocínio lógico, criatividade, capacidade de abstração, capacidade para identificar, analisar, projetar soluções de problemas, a capacidade de comunicação oral e a capacidade de trabalhar em grupo
OTR	Outros	

PROGRAMA:

Tópico	Procedimento Didático	Número de aulas	Início
Introdução	AEX	2	13/03/2018
Representação de Dados: inteiros não negativos	AEX,PRA	4	16/03/2018

Representação de Dados: manipulação de bits / Atividade Supervisionada	AEX,PRA,ED	2	23/03/2018
Representação de Dados: inteiros com sinal / Atividade Supervisionada	AEX,PRA,ED	4	27/03/2018
Representação de Dados: array e estrutura / Atividade Supervisionada	AEX,PRA,ED	2	06/04/2018
Assembly: introdução, operações aritméticas e lógicas / Atividade Supervisionada	AEX,PRA,ED	2	10/04/2018
Assembly: mecanismos de controle / Atividade Supervisionada	AEX,PRA,ED	4	13/04/2018
Revisão	OTR	2	20/04/2018
Prova 1	OTR	2	24/04/2018
Assembly: chamada de funções, passagem de parâmetros e registro de ativação / Atividade Supervisionada	AEX,PRA,ED	10	27/04/2018
Assembly: variáveis locais e registro de ativação / Atividade Supervisionada	AEX,PRA,ED	4	28/05/2018
Revisão	OTR	2	29/05/2018
Prova 2	OTR	2	05/06/2018
Ponto flutuante: representação IEEE 754	AEX,PRA	4	08/06/2018
Assembly: instruções de ponto flutuante / Atividade Supervisionada	AEX,PRA,ED	6	15/06/2018
Interrupções, Exceções e Chamadas ao SO / Atividade Supervisionada	AEX,PRA,ED	2	26/06/2018
Ligação e Relocação de Programas / Atividade Supervisionada	AEX,PRA,ED	6	29/06/2018
Revisão	OTR	2	10/07/2018
Prova 3	OTR	2	13/07/2018
Total	–	64	–

CRITÉRIO DE AVALIAÇÃO:

A avaliação da aprendizagem será feita por meio de três avaliações escritas.

- P1: Avaliação escrita (10 pontos)
- P2: Avaliação escrita (10 pontos)
- P3: Avaliação escrita (10 pontos)

A nota final (NF) do estudante será calculada da seguinte forma:

- $NF = (P1 + P2 + P3) / 3$

Datas das avaliações:

- P1: 24/05/2018
- P2: 05/06/2018
- P3: 13/07/2018

Observações:

- Será atribuída a nota 0 (zero) a qualquer avaliação não realizada.
- O estudante só será aprovado se NotaFinal $\geq 6,0$ e frequência $\geq 48h$.

BIBLIOGRAFIA BÁSICA:

- BRYANT, R.; O'HALLARON, D.; Computer Systems, A programmer's perspective, 2nd edition, Prentice Hall. 2011.
- DUNTEMANN, J.; Assembly Language Step-by-Step: Programming with Linux; 3rd. Edition, Wiley, 1999.
- WARFORD, J. S.; Computer Systems, 4th edition, Jones and Bartlett Publishers, 2010.

BIBLIOGRAFIA COMPLEMENTAR:

- CARTER., P. A., PC Assembly Language, 2006, disponível em: <http://www.drpaulcarter.com/pcasm/>
- CARPINELLI, J. D., Computer systems organization & architecture, Addison-Wesley, 2001.
- HENNESSY, J. L., PATTERSON, D. A.; Computer organization and design : the hardware / software interface, 2nd edition, Morgan Kaufmann Publishers, 1998.
- LOVE, R., Linux system programming, O'Reilly, 2007.
- KERRISK, M. The Linux Programming Interface - A Linux and Unix System Programming Handbook, No Starch Press., 2010.