Insper

Sistemas Hardware-Software

Aula 1 – Introdução + Inteiros na CPU

Ciência da Computação

Carlos Menezes <carlosedm@insper.edu.br>

Professor

- Carlos Eduardo Dantas de Menezes

Aulas

- Aulas
  - Quartas, 15h45 às 17h45
  - Sextas, 15h45 às 17h45

Atendimento

- Presencial: Sala 513
  - Sextas, 14h15 às 15h45

Hoje

- Resumo rápido do curso
- Inteiros na CPU

Critérios para Avaliação

Exercícios práticos (atividades e labs)

- Série de exercícios práticos de implementação
- Complexidade crescente
- Testes automatizados quando possível
  - Facilitar correção
  - Criar espaços para conversar da matéria

Exercícios práticos (entrega)

- Github classroom
  - Testes automatizados para alguns exercícios
  - Ver link e tutorial em **Conteúdos** (Blackboard) para cadastro

Avaliação

- Média Final (MF) se cumpridas as condições:

NS = 0,10 A + 0,20 PI + 0,30 PF + 0,40 L

NC = 0,10 A + 0,20 PI + 0,25 PF + 0,40 L + 0,05 C

MF = max(NS, NC)

A: Atividades (atv)
PI: prova intermediária
PF: prova final
L: laboratórios (labs)
C: prova mutirão C

- Média Final (MF) se NÃO cumpridas as condições:

MF = min(A, PI, PF, L, C)

- Condições:

L >= 5
PI e PF >= 4
((PI + PF) / 2) >= 4,5

Avaliação (DELTA provas)

Se (PI < 4 E PF >= 5) OU (PI >= 5 E PF < 4):

1. Aluno faz uma nova prova PD no dia da SUB relativa a avaliação em que tirou nota menor que 4.

2. Critério de barreira de provas é cumprido se PD >= 5.

Ferramentas

- GCC 9.3 (ou superior) -- C99
- Linux (Preferencialmente ubuntu 22.04)
- PC x86-64

Não há suporte a outros sistemas. Instalem direto ou usem uma VM. Se usar VM, veja se funciona com proctorio.

Resumo do curso

Objetivo de Sistemas Hardware-Software

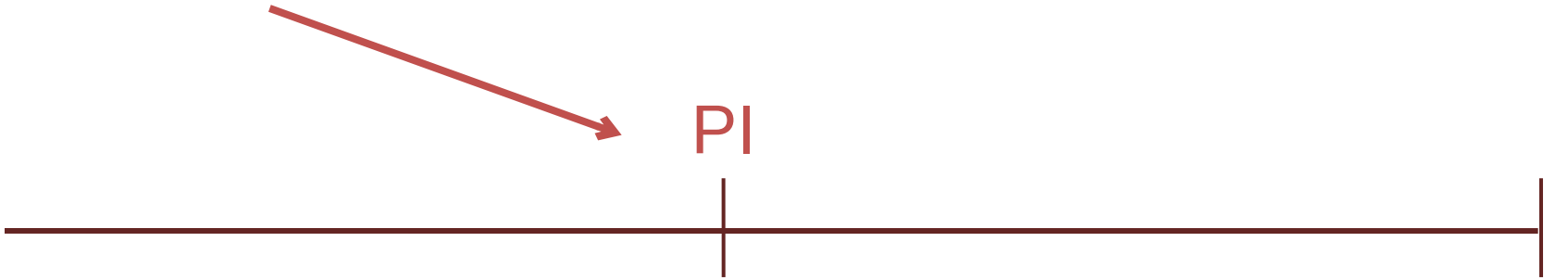
Entender como um programa roda em um PC

- Representação de dados na memória
- Linguagem Assembly x86 (processadores Intel e AMD)
- Sistemas Operacionais (Linux)
  - programas, processos
  - entrada/saída

Visão geral do curso

Linguagem de máquina

- Arquitetura x86
- Compilação
- Linguagem C



- Visão geral de um sistema
- Programas, processos
- Entrada/saída
- Sistema de arquivos

Sistemas Operacionais

Aula!

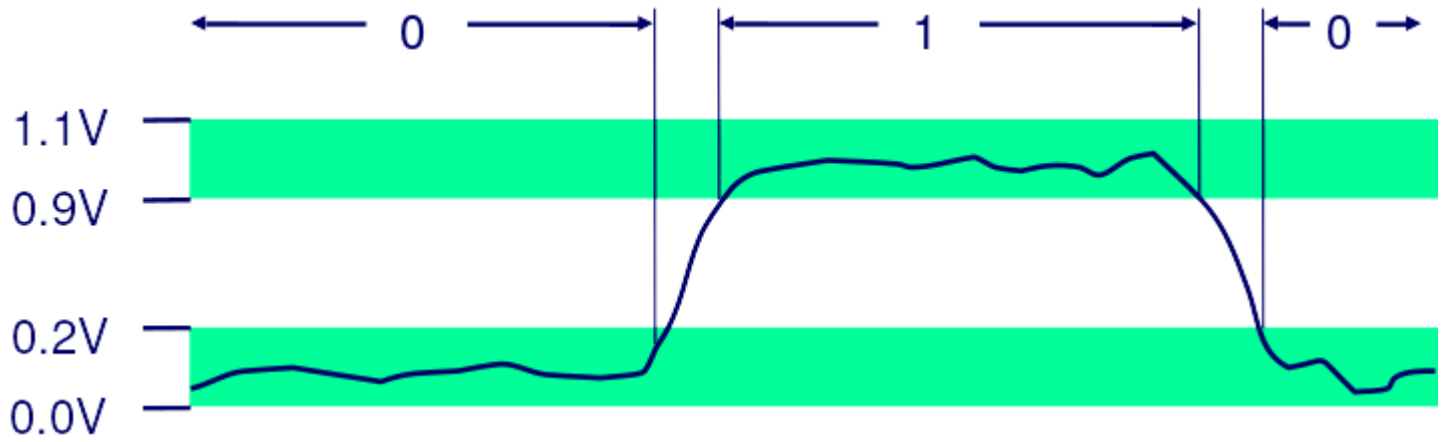
O que é isto?!

480

Representação de inteiros na CPU

Bits e Bytes

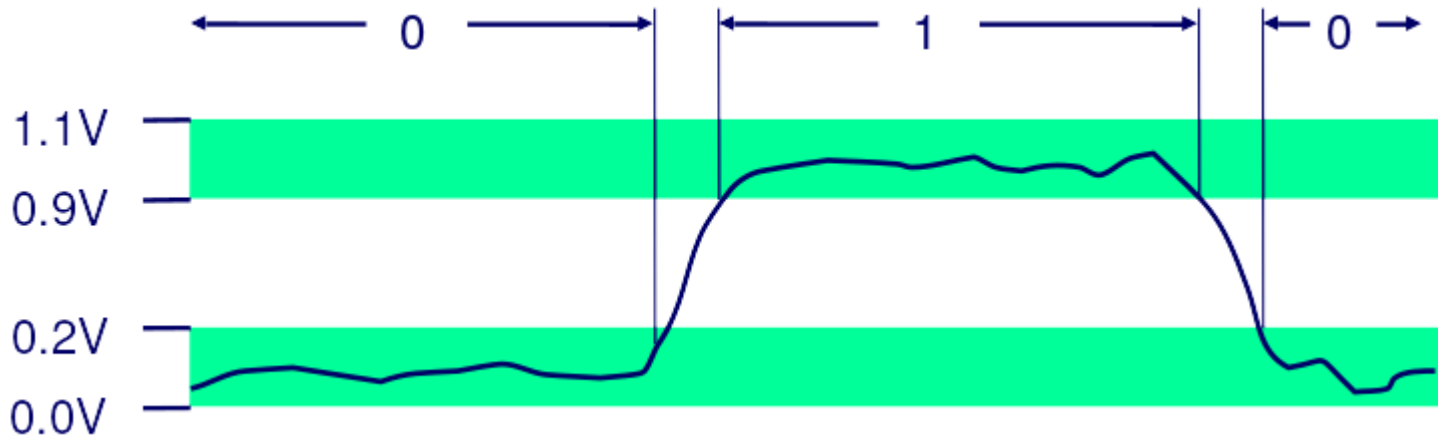
Informação é codificada como sequência de 0 e 1



- Inteiros, Strings, Números reais
- Instruções da CPU, Endereços, etc

Bits e Bytes

Informação é codificada como sequência de 0 e 1



- Inteiros, Strings, Números reais
- Instruções da CPU, Endereços, etc

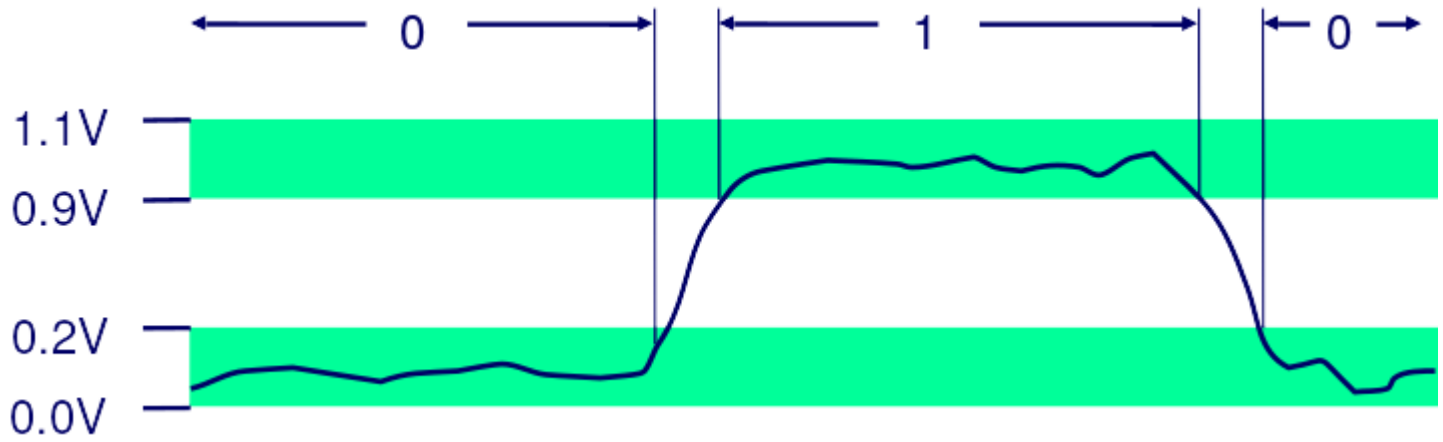
Não é possível distinguir conteúdo a partir de uma sequência de bits

Insper

Bits e Bytes

Agrupamos 8 bits em 1 byte

Informação é codificada como sequência de 0 e 1



Não é possível distinguir conteúdo a partir de uma sequência de bits

- Inteiros, Strings, Números reais
- Instruções da CPU, Endereços, etc

Insper

Inteiros (decimal)

Número **9153**

Inteiros (decimal)

Número **9153**

$$9000 + 100 + 50 + 3 = \mathbf{9}x10^{3} + \mathbf{1}x10^{2} + \mathbf{5}x10^{1} + \mathbf{3}x10^{0}$$

1. Cada dígito multiplica uma potência de 10
2. O dígito **mais significativo** é **9** (multiplica a maior potência)
3. O dígito **menos significativo** é **3** (multiplica a menor potência)

Inteiros (binário)

Número **1010011** (base 2)

Inteiros (binário)

Número **1010011** (base 2)

$$2^6 + 2^4 + 2^1 + 2^0 = \mathbf{83} \text{ (base 10)}$$

1. Cada dígito multiplica uma potência de 2
2. O dígito mais significativo é 1 (multiplica a maior potência)
3. O dígito menos significativo é 0 (multiplica a menor potência)

Conversão Binário -> Decimal: Exercício

Converta o número abaixo para decimal

1100 0010

Conversão Decimal -> Binário

Fazemos agora o caminho inverso: dividimos sucessivamente por 2 e guardamos o resto

75 (base 10)

Conversão Decimal -> Binário: Exercício

Agora é sua vez:

165

Conversão Decimal -> Binário

Forma bônus:

Arquitetura de computadores

- Todo dado tem tamanho **fixo.**
- Um inteiro pode ter os seguintes tamanhos:

Tamanho em bytes	Tipo em C	Capacidade
1	char	
2	short	
4	int	
8	long	

Arquitetura de computadores

- Todo dado tem tamanho **fixo.**
- Um inteiro pode ter os seguintes tamanhos:

Tamanho em bytes	Tipo em C	Capacidade
1	char	256
2	short	65536
4	int	2^{32}
8	long	2^{64}

Inteiros sem sinal

Representação para números positivos somente (modificador unsigned)

Tamanho em bytes	Tipo em C	Menor número	Maior Número
1	char	0	
2	short	0	
4	int	0	
8	long	0	

Inteiros sem sinal

Representação para números positivos somente (modificador unsigned)

Tamanho em bytes	Tipo em C	Menor número	Maior Número
1	char	0	255
2	short	0	65535
4	int	0	$2^{32} - 1$
8	long	0	$2^{64} - 1$

Inteiros com sinal (Complemento de dois)

Dado um inteiro $\mathbf{b_2}$ com **w** bits, seu valor em decimal é

$$b_{10} = -2^{w-1} b_{w-1} + \sum_{i=0}^{w-2} 2^i b_i$$

1. Somamos todos os bits normalmente
2. Menos o último, que ao invés de somar **subtrai**

Inteiros com sinal (Complemento de dois)

Na notação em complemento de dois, o dígito mais à esquerda é utilizado para representar o sinal: em um número negativo, ele é 1, e num número positivo, é 0.

Um algoritmo para representar um número negativo em binário:

1) Escreva o módulo (positivo) do valor em binário.

2) Inverta todos os bits.

3) Some 1 ao resultado.

Inteiros com e sem sinal

Qual o valor de 0100 0101 (base2) em base 10?

Sem sinal:

Com sinal:

Inteiros com e sem sinal

Qual o valor de 0100 0101 (base2) em base 10?

Sem sinal:

$2^6 + 2^2 + 2^0 =$ **71** (base 10)

Com sinal:

$2^6 + 2^2 + 2^0 =$ **+71** (base 10)

Inteiros com e sem sinal - Exercício

Qual o valor de 0101 1010 (base2) em base 10?

Sem sinal:

Com sinal:

Inteiros com e sem sinal

Qual o valor de 11 0001 (base 2)?

Sem sinal:

Com sinal:

Inteiros com e sem sinal

Qual o valor de 11 0001 (base 2)?

Sem sinal:

$2^5 + 2^4 + 2^0 = 49$ (base 10)

Com sinal:

$-2^5 + 2^4 + 2^0 = -32 + 17 = -15$ (base 10)

Inteiros com e sem sinal – Exercício

Qual o valor de 1 0101 0001 (base 2)?

Sem sinal:

Com sinal:

Hexadecimal

Os dois números abaixo são o mesmo? Se não qual o bit diferente?

1001110011101110

1001110111101110

Hexadecimal

Os dois números abaixo são o mesmo?

0x9CEE

0x9DEE

Hexadecimal

Os dois números abaixo são o mesmo?

0x9CEE

0x9DEE

Objetivo: facilitar a leitura de números binários

Hexadecimal

Os dois números abaixo são o mesmo?

0x9CEE

0x9DEE

Ideia:

- agrupar 4 em 4 bits em um dígito que vai de 0 a 15
- letras para os dígitos maiores que 10

Hexadecimal

Binário	Hexa	Binário	Hexa
0000	0x0	1000	0x8
0001	0x1	1001	0x9
0010	0x2	1010	0xA
0011	0x3	1011	0xB
0100	0x4	1100	0xC
0101	0x5	1101	0xD
0110	0x6	1110	0xE
0111	0x7	1111	0xF

Exercício

Converta para binário: 0xDE9 (base 16)

Converta para hexadecimal: 1100 1110 0011 1010 (base 2)

Exercício

Converta para binário: 0xDE9 (base 16)

1101 1110 1001 (base 2)

Converta para hexadecimal: 1100 1110 0011 1010 (base 2)

0xCE3A (base 16)

Conversões de tipos

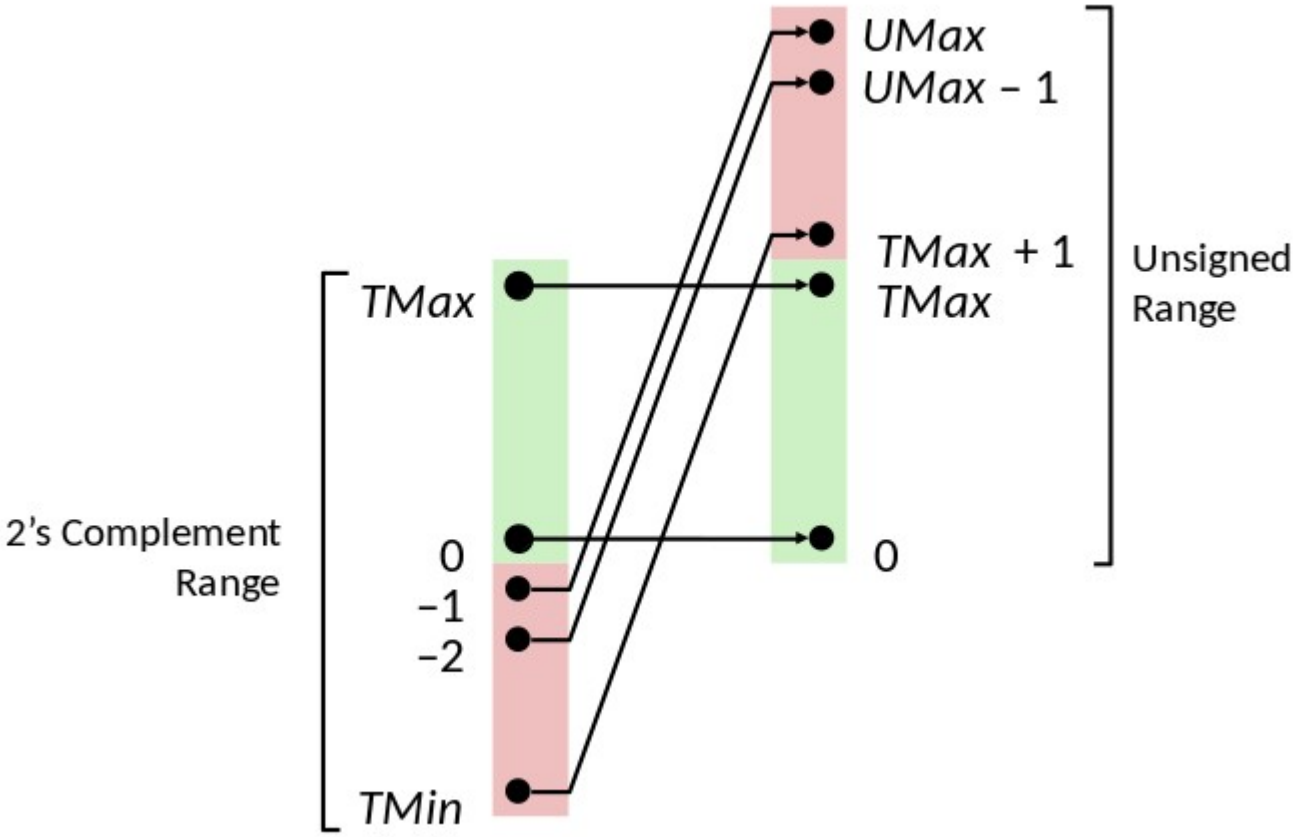
Conversões de tipos inteiros

Duas regras:

1. O valor é mantido quando convertemos de um tipo menor para um tipo maior
   - char -> int

2. A conversão de um tipo maior para um tipo menor é feita pegando o X bits menos significativos
   - int -> char pega os 8 bits menos significativos, o restante é descartado

Conversões de tipos inteiros - sinal



Insper

Atividade prática

Conversão de números: bases e sinal

1. rodar programa bases_e_sinais
2. colocar sua solução em solucao.txt
3. verificar se tudo está ok rodando

```
./bases_e_sinais < solucao.txt
```

Atividade Extra (Não será cobrada)

Atividade extra para os curiosos!

Pesquise como o computador representa números reais. Qual o padrão utilizado?

Git

https://insper.github.io/SistemasHardwareSoftwareBCC/

https://github.com/Insper/SistemasHardwareSoftwareBCC

Insper

www.insper.edu.br