

NOME:

Nº:

**Notas:**

1. Para cada uma das **5 questões de resposta satisfatória obrigatória**, numeradas de **1 a 5**, são-lhe oferecidas pelo menos 2 hipóteses para responder e/ou comentar; para cada um destas deverá optar por responder a **apenas uma** delas.
2. Para cada uma das hipóteses que optar, deverá apresentar a **justificação da solução**, incluindo o raciocínio ou os cálculos que efetuar.
3. **Não são permitidas:** (i) máquinas de calcular e (ii) notas auxiliares de memória.
4. Correção de cada questão: não-satisfaz (**0**), satisfaz com erros (**0.8**), certa com falhas (**1.0**) e completamente certa (**1.2**).

1.

- a) Uma cadeia de hotelaria tem 7 hotéis espalhados pelo mundo. Cada quarto é identificado por uma sequência de 3 dígitos que identificam o hotel, o andar onde o quarto se localiza e o número do quarto dentro desse hotel. Nenhum dos hotéis tem mais de 28 andares, sendo que até 5 destes andares podem ser subterrâneos (assuma portanto que nenhum hotel tem mais de 23 andares não subterrâneos); nenhum dos andares tem mais de 65 quartos. Proponha um formato para representar o identificador de um quarto em binário, usando o menor número de bits possível. Os andares subterrâneos devem ser representados com um número negativo. Ilustre com o caso do hotel número 3, andar -2 e quarto 31.
- b) Uma empresa desenvolveu um protocolo de comunicação que usa apenas os 10 primeiros caracteres do alfabeto e ainda o carácter “;” para separar palavras. Proponha um código binário para representar estes caracteres e ilustre essa representação para a *string* “JA;HHH”.

2. Considere um PC com o processador IA-20, semelhante ao IA-32 (*little endian* e onde inteiros são representados em 20-bits, em complemento para 2) e com células de memória de 4 bits.

- a) Considerando que o registo `%axs` contém um valor inteiro representado em hexadecimal por `0x4c800`, **indique em decimal**, o valor que está lá guardado.
- b) Considere a instrução `mov $-520, 8(%axs)`. **Mostre em binário** o valor que vai ficar armazenado na célula de memória em `0x4c80a`, após execução dessa instrução.

3. Considere o mesmo IA-20 da questão anterior, que suporta uma representação de números reais usando uma versão reduzida da norma IEEE 754 com 20 bits (7 bits para o expoente em excesso de  $2^{(7-1)} - 1$ , 12 para a mantissa e 1 para o sinal; não esquecer os casos de exceção).

- a) **Mostre, em binário**, o conteúdo de um registo FP de 20 bits contendo o valor  $1/10$ .
- b) **Indique, em decimal**, que valor se encontra armazenado num destes registos FP de 20 bits, se o seu conteúdo for `0x80080`.

4. Considere o estado parcial de um PC com um processador IA-32 ilustrado abaixo. Considere também que se trata do excerto de código final de uma função, estando mantido na *stack* o quadro de activação dessa função.

Registos	Memória (código)	Memória (dados)
<code>%eip = 0x00004051</code>	<code>0x4050 inc %eax</code>	<code>0x06FFC: 0xAA 0x00 0x00 0x01</code>
<code>%ebp = 0x00007000</code>	<code>0x4051 addl %eax, 12(%ebp)</code>	<code>0x07000: 0x10 0x70 0x00 0x00</code>
<code>%eax = 0x00000100</code>	<code>0x4054 popl %eax</code>	<code>0x07004: 0xb5 0x40 0x00 0x00</code>
<code>%ebx = 0xFFFFFFFF</code>	<code>0x4055 movl %ebp, %esp</code>	<code>0x07008: 0x20 0x00 0x11 0x00</code>
<code>%ecx = 0x00000001</code>	<code>0x4056 popl %ebp</code>	<code>0x0700C: 0x01 0x00 0xFF 0x00</code>
<code>%esp = 0x00006FFC</code>	<code>0x4057 ret</code>	

- a) **Indique, justificando**, o valor do registo `%eax` no final da execução desta função.
- b) **Identifique, justificando**, a veracidade da seguinte afirmação: “O *frame pointer* da função que invocou a função acima aponta para o endereço `0x10700000`”.

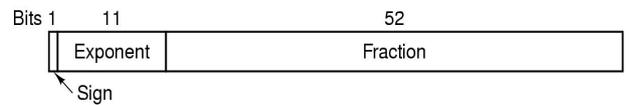
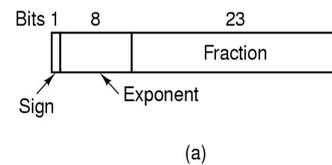
5. Ainda relativamente ao estado parcial do computador da questão anterior, considere a execução da instrução que se encontra na memória em 0x4054: `popl %eax`

- a) **Apresente**, por ordem cronológica, toda a informação que circula apenas no barramento de endereços.
- b) **Indique** todos os registos e todas as células de memória que foram modificados com a execução desta instrução, bem como os respectivos valores antes e depois da execução da instrução.

**Notas de apoio (norma IEEE 754)**

Normalized	±	0 < Exp < Max	Any bit pattern
Denormalized	±	0	Any nonzero bit pattern
Zero	±	0	0
Infinity	±	1 1 1...1	0
Not a number	±	1 1 1...1	Any nonzero bit pattern

↙ Sign bit



Valor decimal de um fp em binário:

precisão simples, normalizado:

$$V = (-1)^S * (1.F) * 2^{E-127}$$

precisão simples, desnormalizado:

$$V = (-1)^S * (0.F) * 2^{-126}$$

precisão dupla, normalizado:

$$V = (-1)^S * (1.F) * 2^{E-1023}$$

precisão dupla, desnormalizado:

$$V = (-1)^S * (0.F) * 2^{-1022}$$