

```

1  Questão 1
2
3  f = 150 MHz = 150 * 10^6 Hz = 150 * 10^6 ciclos / segundo
4
5  a)
6  f = 1 / t
7  f = 150 * 10^6 ciclos / segundo
8  f = 150 * 10^6 * 10^-3 ciclos / milisegundo
9  f = 15 * 10^4 ciclos / milisegundo
10
11 t = 1 / (15 * 10^4) milisegundos / ciclo
12 t = (1 / 15) * 10^-4 milisegundos / ciclo
13 t = 0,000006666666666 milisegundos / ciclo
14
15
16 b)
17 f = 1 / t
18 f = 150 * 10^6 ciclos / segundo
19 f = 150 * 10^6 * 10^-6 ciclos / microsegundo
20 f = 15 * 10^1 ciclos / microsegundo
21
22 t = 1 / (15 * 10^1) microsegundos / ciclo
23 t = (1 / 15) * 10^-1 microsegundos / ciclo
24 t = 0,006666666666666 microsegundos / ciclo
25
26
27 c)
28 f = 1 / t
29 f = 150 * 10^6 ciclos / segundo
30 f = 150 * 10^6 * 10^-9 ciclos / nanosegundo
31 f = 15 * 10^-2 ciclos / nanosegundo
32
33 t = 1 / (15 * 10^-2) nanosegundos / ciclo
34 t = (1 / 15) * 10^2 nanosegundos / ciclo
35 t = 6,666666666666666 nanosegundos / ciclo
36
37 -----
38
39 Questão 2
40
41 t = 100 nanosegundos / ciclo
42 t = 10^2 nanosegundos / ciclo
43
44 t = 10^2 * 10^-3 microsegundos / ciclo
45 t = 10^-1 microsegundos / ciclo
46
47 t = 10^2 * 10^-6 milisegundos / ciclo
48 t = 10^-4 milisegundos / ciclo
49
50 a)
51 f = 1 / t
52 t = 10^-4 milisegundos / ciclo
53 f = 10^4 ciclos / milisegundo = 10^4 KHz
54
55 b)
56 f = 1 / t
57 t = 10^-1 microsegundos / ciclo
58 f = 10^1 ciclos / microsegundo = 10^1 MHz
59
60 c)
61 f = 1 / t
62 t = 10^2 nanosegundos / ciclo
63 f = 10^-2 ciclos / nanosegundo = 10^-2 GHz
64
65 -----
66
67 Questão 3
68
69 p -> propaga o vai 1 -> or das entradas
70 g -> gera o vai 1 -> and das entradas
71
72 a)
73 FACA + BABA

```

```

74
75         ponmlkjihgfedcba
76
77 FACA  1111101011001010
78 BABA  1011101010111010
79
80 p      1111101011111010
81 g      1011101010001010
82
83 Sim, terá vai 1 no final, pois os bits da coluna p geram e propagam o vai 1 final
84
85 b)
86 3AF1 + 45EA
87
88         ponmlkjihgfedcba
89
90 3AF1  0011101011110001
91 45EA  0100010111101010
92
93 p      0111111111111011
94 g      0000000011100000
95
96 Não terá vai 1 final, pois os bits da coluna p não o propagarão
97
98 -----
99
100 Questão 4
101
102 Tendo em mente que os números estão na base 2, e que existem o multiplicando e o
multiplicador,
103 o que o algoritmo faz é:
104
105 1º - inicia o produto como 0
106 2º - soma o multiplicando ao produto caso o bit mais à direita do multiplicador seja 1
107 3º - desloca o multiplicador uma vez para a direita
108 4º - desloca o produto uma vez para a direita
109 5º - finaliza se todos os bits do multiplicador tiverem sido usados, caso contrário,
volta a 2º
110
111 Exemplo, 3 * 5:
112
113 1 ->
114 multiplicando = 3 = 0011
115 multiplicador = 5 = 0101
116 produto = 0000 0000
117
118 2 ->
119 multiplicando = 3 = 0011
120 multiplicador = 5 = 0101 (o bit mais à direita do multiplicador é 1, então soma-se o
multiplicando ao produto)
121 produto = 0011 0000
122
123 3 -> (bits usados do multiplicador = 1)
124 multiplicando = 3 = 0011
125 multiplicador = 5 = 0010 (deslocar o multiplicador uma vez para a direita)
126 produto = 0001 1000 (deslocar o produto uma vez para a direita)
127
128 4 ->
129 multiplicando = 3 = 0011
130 multiplicador = 5 = 0010 (o bit mais à direita do multiplicador é 0, então não se
faz nada)
131 produto = 0001 1000
132
133 5 -> (bits usados do multiplicador = 2)
134 multiplicando = 3 = 0011
135 multiplicador = 5 = 0001 (deslocar o multiplicador uma vez para a direita)
136 produto = 0000 1100 (deslocar o produto uma vez para a direita)
137
138 6 ->
139 multiplicando = 3 = 0011
140 multiplicador = 5 = 0001 (o bit mais à direita do multiplicador é 1, então soma-se o
multiplicando ao produto)
141 produto = 0011 1100

```

[illegible]

[illegible]

```

274
275 d)  $-4096 = -1000_{16} = -1000000000000_{2} = -1.000000000000_{2} * 2^{12}$ 
276
277  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_rep} = \text{exp\_real} + \text{bias} =$ 
278  $12 + 127 = 139$ 
279
280 s      expoente      mantissa
281 0      10001011      000000000000000000000000 -> normalizado
282 leading bit = 1
283
284 e)  $-3.25 = -3.4_{16} = -11.01_{2} = -1.101_{2} * 2^1$ 
285
286  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_rep} = \text{exp\_real} + \text{bias} =$ 
287  $1 + 127 = 128$ 
288
289 s      expoente      mantissa
290 0      10000000      101000000000000000000000 -> normalizado
291 leading bit = 1
292 -----
293
294 Questão 10
295
296 a) 0 10000001 011000000000000000000000
297
298  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_real} = \text{exp\_rep} - \text{bias} =$ 
299  $129 - 127 = 2$ 
300
301  $M = \text{mantissa} + ( \text{normalizado} ? 1 : 0 ) = 0.375 + 1 = 1.375$ 
302
303  $\text{valor} = (-1)^s * M * 2^{\text{exp\_real}} = 1 * 1.375 * 2^2 = 5.5$ 
304
305 b) 1 10000001 000100000000000000000000
306
307  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_real} = \text{exp\_rep} - \text{bias} =$ 
308  $129 - 127 = 2$ 
309
310  $M = \text{mantissa} + ( \text{normalizado} ? 1 : 0 ) = 0.0625 + 1 = 1.0625$ 
311
312  $\text{valor} = (-1)^s * M * 2^{\text{exp\_real}} = -1 * 1.0625 * 2^2 = -4.25$ 
313
314 c) 1 10000000 000000000000000000000000
315
316  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_real} = \text{exp\_rep} - \text{bias} =$ 
317  $128 - 127 = 1$ 
318
319  $M = \text{mantissa} + ( \text{normalizado} ? 1 : 0 ) = 0.0 + 1 = 1$ 
320
321  $\text{valor} = (-1)^s * M * 2^{\text{exp\_real}} = -1 * 1 * 2^1 = -2$ 
322
323 d) 0 00000001 010110000000000000000000
324
325  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_real} = \text{exp\_rep} - \text{bias} =$ 
326  $1 - 127 = -126$ 
327
328  $M = \text{mantissa} + ( \text{normalizado} ? 1 : 0 ) = 0.34375 + 1 = 1.34375$ 
329
330  $\text{valor} = (-1)^s * M * 2^{\text{exp\_real}} = 1 * 1.34375 * 2^{(-126)} = 1.5795 * 10^{(-38)}$ 
331
332 -----
333
334 Questão 11
335
336 a) 0x41202000 = 0 10000010 010000000100000000000000 -> normalizado
337
338  $(IEEE\ 754 - 1/8/23), \text{bias} = 2^{(n-1)} - 1 = 2^7 - 1 = 127, \text{exp\_real} = \text{exp\_rep} - \text{bias} =$ 
339  $130 - 127 = 3$ 

```

```

340 M = mantissa + ( normalizado ? 1 : 0 ) = 0.2509 + 1 = 1.2509
341
342 valor = (-1)^s * M * 2 ^ exp_real = 1 * 1.2509 * 2 ^ 3 = 10.0072
343
344
345 b) 0x00000000 = 0 00000000 000000000000000000000000 -> não normalizado, mantissa == 0
-> valor = 0
346
347 (IEEE 754 - 1/8/23), bias = 2^(n-1) - 1 = 2^7 - 1 = 127
348
349 M = mantissa + ( normalizado ? 1 : 0 ) = 0.0 + 0 = 0.0
350
351 valor = (-1)^s * M * 2 ^ exp_real = 1 * 0.0 * 2 ^ 0 = 0.0
352
353
354 c) 0x42E80000 = 0 10000101 110100000000000000000000 -> normalizado
355
356 (IEEE 754 - 1/8/23), bias = 2^(n-1) - 1 = 2^7 - 1 = 127, exp_real = exp_rep - bias =
133 - 127 = 6
357
358 M = mantissa + ( normalizado ? 1 : 0 ) = 0.8125 + 1 = 1.8125
359
360 valor = (-1)^s * M * 2 ^ exp_real = 1 * 1.8125 * 2 ^ 6 = 116
361
362
363 d) 0xC1DA0000 = 1 10000011 101101000000000000000000 -> normalizado
364
365 (IEEE 754 - 1/8/23), bias = 2^(n-1) - 1 = 2^7 - 1 = 127, exp_real = exp_rep - bias =
131 - 127 = 4
366
367 M = mantissa + ( normalizado ? 1 : 0 ) = 0.703125 + 1 = 1.703125
368
369 valor = (-1)^s * M * 2 ^ exp_real = 1 * 1.703125 * 2 ^ 4 = 27.25
370
371
372 e) 0xFFFFFFFF = 1 11111111 111111111111111111111111 -> não normalizado, mantissa != 0
-> valor =  $\sqrt{(-1)}$ ,  $\infty - \infty$ , etc (NaN)
373
374 (IEEE 754 - 1/8/23), bias = 2^(n-1) - 1 = 2^7 - 1 = 127
375
376 M = mantissa + ( normalizado ? 1 : 0 ) = ???
377
378 valor = (-1)^s * M * 2 ^ exp_real = Not a Number
379
380
381 -----
382
383 Questão 12
384
385 -----
386
387 Questão 13
388
389 b) D57F0000 = 1 10101010 111111100000000000000000
390 e) D5FFE000 = 1 10101011 111111111000000000000000
391 c) 5F7FF800 = 0 10111110 111111111110000000000000
392 a) 7F7FF800 = 0 11111110 111111111110000000000000
393 d) 7F800000 = 0 11111111 000000000000000000000000
394
395 -----
396
397 Questão 14
398
399 a)  $1.12 * 10^2 = 112 = 1110000_{\{2\}} = 1.11 * 2^6$ 
400
401 (IEEE 754 - 1/4/3), bias = 2^(n-1) - 1 = 2^3 - 1 = 7, exp_rep = exp_real + bias = 6
+ 7 = 13
402
403 s      expoent mantissa
404 0      1101      110 -> normalizado
405 leading bit = 1
406 (Não houve nenhuma perda de precisão na transferência do 1º número para IEEE-754)
407

```

```

408
409  $2.24 * 10^{(-1)} = 0.00111001010110000001_{(2)} = 1.11001010110000001 * 2^{(-3)}$ 
410
411 (IEEE 754 - 1/4/3), bias =  $2^{(n-1)} - 1 = 2^3 - 1 = 7$ , exp_rep = exp_real + bias = -3 + 7 = 4
412
413 s    expoent mantissa
414 0    0100    110 -> normalizado
415 leading bit = 1
416 (Houve perda de precisão, novo valor =  $1.11 * 2^{(-3)} = 0.21875$ )
417
418 Soma com erro =  $112 + 0.21875 = 112.21875$ 
419 Soma sem erro =  $112 + 0.224 = 112.224$ 
420 Erro relativo =  $1 - (Soma\ com\ erro / Soma\ sem\ erro) = 1 - 0.99995 = 0.00005 = 0.005\%$ 
421 Erro absoluto =  $112.224 - 112.21875 = 0.00525$ 
422
423 Produto com erro =  $112 * 0.21875 = 24.5$ 
424 Produto sem erro =  $112 * 0.224 = 25.088$ 
425 Erro relativo =  $1 - (Produto\ com\ erro / Produto\ sem\ erro) = 1 - 0.9765 = 0.0234 = 2.34\%$ 
426 Erro absoluto =  $25.088 - 24.5 = 0.588$ 
427
428
429 b)  $1.12 * 10^2 = 112 = 1110000_{(2)} = 1.11 * 2^6$ 
430
431 (IEEE 754 - 1/3/4), bias =  $2^{(n-1)} - 1 = 2^2 - 1 = 3$ , exp_rep = exp_real + bias = 6 + 3 = 9 = 1001
432
433 s    exp mantissa
434 0    100 1100 -> normalizado
435 leading bit = 1
436 (Houve perda de precisão, novo valor =  $1.11 * 2^1 = 11.1 = 3.5$ )
437
438
439  $2.24 * 10^{(-1)} = 0.00111001010110000001_{(2)} = 1.11001010110000001 * 2^{(-3)}$ 
440
441 (IEEE 754 - 1/3/4), bias =  $2^{(n-1)} - 1 = 2^2 - 1 = 3$ , exp_rep = exp_real + bias = -3 + 3 = 0
442
443 s    exp mantissa
444 0    000 1100 -> não normalizado, mantissa != 0 -> exp = -bias + 1 = -2
445 leading bit = 1
446 (Houve perda de precisão, novo valor =  $1.11 * 2^{(-2)} = 0.4375$ )
447
448 Soma com erro =  $3.5 + 0.4375 = 3.9375$ 
449 Soma sem erro =  $112 + 0.224 = 112.224$ 
450 Erro relativo =  $1 - (Soma\ com\ erro / Soma\ sem\ erro) = 1 - 0.03508 = 0.9649 = 96.49\%$ 
451 Erro absoluto =  $112.224 - 3.9375 = 108.2865$ 
452
453 Produto com erro =  $3.5 * 0.4375 = 1.53125$ 
454 Produto sem erro =  $112 * 0.224 = 25.088$ 
455 Erro relativo =  $1 - (Produto\ com\ erro / Produto\ sem\ erro) = 1 - 0.0610 = 0.9389 = 93.89\%$ 
456 Erro absoluto =  $25.088 - 1.53125 = 23.55675$ 
457
458 -----
459
460 Questão 15
461
462 -----
463
464 Questão 16
465
466 vel_mem_princ = x -> vel_mem_cache = 5x
467 supondo que o tempo gasto sem melhoria é  $10t = t + 9t$ , com a melhoria, o tempo passa a ser de
468  $t + 9t/5 = 14t/5$ , portanto, o speed up é igual a  $10t / (14t / 5) = 10t * 5 / 14t = 50 / 14 = 25 / 7 = 3 + 4 / 7 = 3.57$ 
469
470 -----
471
472 Questão 17
473

```

474 Supondo que o tempo gasto sem a melhoria seja $10t$:

475

476 a) tempo gasto com a melhoria = $5t + 5t / 5 = 6t \rightarrow \text{speed up} = 10t / 6t = 10 / 6$
 $= 5 / 3 = 1 + 2 / 3 = 1.444444...$

477

478 b) $\text{speed up} = 2 = \text{tempo_sem_melhoria} / \text{tempo_com_melhoria} = 10t / \text{tempo_com_melhoria}$
 $\rightarrow \text{tempo_com_melhoria} = 5t$, supondo
 479 que a utilização do coprocessador seja igual a k , a utilização do processador será $1 - k$, daí posso montar a relação:
 480 $(1 - k) * 10t + k * 10t / 5 = 5t \rightarrow$ dividindo tudo por $5t \rightarrow (1 - k) * 2 + k * 2 / 5$
 $= 1 \rightarrow$
 481 $2 - 2k + 2k / 5 = 1 \rightarrow 1 = 2k - 2k / 5 \rightarrow 1 = 4 * 2k / 5 \rightarrow k = 5 / 8 = 62,5\%$

482

483 c) supondo que essa quantidade de vezes seja x , posso montar a relação: $10t / (5t + 5t / x) = 2.5 \rightarrow$
 484 $4t / (5t + 5t / x) = 1 \rightarrow 4t = 5t + 5t / x \rightarrow 0 = t + 5t / x \rightarrow -t = 5t / x \rightarrow x = -5$ (IMPOSSÍVEL)

485 -----

486

487

488 Questão 18

489

490 $\text{tempo_total_com_o_1º_processador} = t$

491 $\text{preço_da_maquina_com_o_1º_processador} = p$

492 1º processador \rightarrow velocidade = v_1 , preço = p_1

493 2º processador \rightarrow velocidade = v_2 , preço = p_2

494 Uso do 2º processador = 50%

495 Relações: $v_2 = 5 * v_1$, $p_2 = 5 * p_1$, $p_1 = 1/3 * p$

496 $\text{speed_up} = t / (t/2 + (t / 2) / 5) = 5 / 3 \approx 1.6$

497 $\text{speed_up_financeiro} = (2p/3 + 1p/3 * 5) / p = 7 / 3 \approx 2.3$

498 Não valhe a pena, pois o speed up financeiro é maior que o speed up de desempenho

499

500 -----

501

502 Questão 19

503

504 $\text{tempo_total_sem_melhoria} = t$

505

506 1º caso:

507 $\text{speed_up1} = t / (4t/5 + (t / 5) / 10) = 50/41 \approx 1.22$

508 2º caso:

509 $\text{speed_up2} = t / (t/2 + (t / 2) / 2) = 4/3 \approx 1.3$

510

511 A melhor alternativa é o 2º caso por ter um speed up maior.

512

513 -----

514

515 Questão 20

516

517 $\text{tempo_total_inicial} = t$

518 $\text{tempo_total_depois_da_melhoria} = t'$

519 $\text{porcentagem_inicial_melhorada} = k$

520 $\text{speed_up} = \text{tempo_sem_melhoria} / \text{tempo_com_melhoria}$

521 $\text{tempo_com_melhoria} = \text{tempo_sem_melhoria} / \text{speed_up}$

522 $\text{speed_up_fração_melhorada} = 10 =$

523 $\text{tempo_sem_melhoria_fração_melhorada} / \text{tempo_com_melhoria_fração_melhorada}$

524

525 Resposta para as duas alternativas:

526 $\text{speed_up} = t / (t * (1 - k) + t * k / 10) = 10 / (10 - 9k)$

527 $\text{speed_up} = t / t' \rightarrow t' = t / \text{speed_up} \rightarrow t' / 2 = t / (2 * \text{speed_up})$

528 $t' / 2$ representa a fração melhorada. Como o fator de melhoramento foi de

529 10, na versão inicial, essa parte era 10 vezes mais lenta, ou seja, gastava

530 10 vezes mais tempo $\rightarrow t * k = (t / (2 * \text{speed_up})) * 10 \rightarrow k = 10 / (2 * \text{speed_up})$

531 $2 * \text{speed_up} = 10 / k \rightarrow \text{speed_up} = 5 / k \rightarrow$

532 $5 / k = 10 / (10 - 9k) \rightarrow k = 10/11 \rightarrow \text{speed_up} = 55/10 = 5.5$

533

534 -----

535

536 Questão 21

537

538 $\text{quantidade_de_ciclos_que_o_programa_gasta_em_A} = 10 * 400 * 10^6 = 4 \text{ Gigaciclos}$

539 $\text{quantidade_de_ciclos_que_o_programa_gasta_em_B} = 4 \text{ Gigaciclos} * 1.2 = 4.8 \text{ Gigaciclos}$

540 $\text{speed_up} = 3 / 2 = \text{tempo_em_A} / \text{tempo_em_B} \rightarrow \text{tempo_em_B} = 20/3 \rightarrow$


```

541  frequência_em_B = 4.8 Gigaciclos / 20/3 segundos = 0.72 GHz = 720MHz
542
543  -----
544
545  Questão 22
546
547  tempo = 1000 * 1 / 100MHz = 1000 / (100 * 10^6) = 10 / 10^6 = 10 microsegundos
548
549  -----
550
551  Questão 23
552
553  tempo_programa1 = 2000 * 5 / 100 MHz = 100 / 10^6 = 100 microsegundos
554  tempo_programa2 = 3000 * 5 / 100 MHz = 150 / 10^6 = 150 microsegundos
555  speed_up = 150 microsegundos / 100 microsegundos = 1.5
556
557  -----
558
559  Questão 24
560
561  CPI_médio = (4 * 60 + 5 * 40) / 100 = 4.4
562  CPI_médio_novo = (3 * 60 + 5 * 40) / 100 = 3.8
563  Como a quantidade de instruções não muda nem a frequência,
564  o speed up é apenas uma relação entre os CPIs:
565  speed_up = 4.4 / 3.8 = 1.15
566
567  -----
568
569  Questão 25
570
571  a) CPI_médio = (4 * 40 + 3 * 30 + 5 * 20 + 6 * 10) / 100 = 4.1
572  b) Mudando apenas a frequência, o speed_up muda da mesma forma -> speed_up = 1.12
573  c) CPI_médio_novo = (2 * 40 + 3 * 30 + 6 * 20 + 6 * 10) / 100 = 3.5
574  Mudando apenas o CPI_médio, a speed_up muda da mesma forma -> speed_up = 4.1 / 3.5 =
575  1.17
576  d) Supondo um total de 100 instruções, as instruções na ALU passariam a ser 20.
577  Dessa forma,
578  as novas porcentagens são:
579  ALU:      20 / 80 = 1/4
580  desvio:   30 / 80 = 3/8
581  memória:  20 / 80 = 1/4
582  outras:   10 / 80 = 1/8
583  CPI_médio_novo = ( 1/4 * 4 + 3/8 * 3 + 1/4 * 5 + 1/8 * 6 ) / 1 = 4.125
584  Como a questão altera a quantidade de instruções e o CPI médio, é necessário
585  calcular o speed up utilizando a fórmula de CPU time:
586  CPU_Time_antigo = 100 * 4.1 * f
587  CPU_Time_novo   = 80 * 4.125 * f
588  speed_up        = (5 * 4.1) / (4 * 4.125) ≈ 1.24
589
590  -----
591
592  Questão 2.1
593
594  A máquina M2 é mais rápida para o programa 1. A máquina M1 é mais rápida para o
595  programa 2.
596  No programa 1, a máquina M2 é mais rápida 10 / 5 vezes = 2 vezes
597  No programa 2, a máquina M1 é mais rápida 4 / 3 vezes ≈ 1.3 vezes
598
599  -----
600
601  Questão 2.2
602
603  velocidade_M1 = 200 Megainstruções / 10 segundos = 20 Megainstruções / segundo = 20
604  * 10^6 instruções / segundo
605  velocidade_M2 = 160 Megainstruções / 5 segundos = 32 Megainstruções / segundo = 32
606  * 10^6 instruções / segundo
607
608  -----
609
610  Questão 2.3
611
612  CPI = (f * t) / IC
613  CPI_M1 = 200 MHz * 10 s / 200 MI = 200Hz * 10 s / 200 I = 10 s * Hz / I = 10 C/I

```

```

609 CPI_M2 = 300 MHz * 10 s / 200 MI = 300Hz * 10 s / 200 I = 15 s * Hz / I = 15 C/I
610
611 -----
612
613 Questão 2.4
614
615 IC = (f * t) / CPI
616 IC_M1 = (200 MHz * 3 s) / 10 C/I = 60 * 10^6 = 60 MI
617 IC_M2 = (300 MHz * 4 s) / 15 C/I = 80 * 10^6 = 80 MI
618
619 -----
620
621 Questão 2.14
622
623 CPU_Time = IC * CPI * tempo_cada_ciclo = IC * CPI / f
624 { CPI, frequência do clock, número de instruções do programa }
625
626 -----
627
628 Questão 2.18
629
630 CPI_MBásica = ( 2 * 40 + 3 * 25 + 3 * 25 + 5 * 10 ) / 100 = 2.8
631 CPI_Mopt      = ( 2 * 40 + 2 * 25 + 3 * 25 + 4 * 10 ) / 100 = 2.45
632
633 -----
634
635 Questão 2.19
636
637 A MBásica realiza 500 Megaciclos / segundo, como cada instruções gasta 2.8 ciclos,
638 a quantidade de instruções por segundo é dada por: (500 MC / s) / 2.8 C/I =
639 178571428.571429 =
640 178.571428571429 MIPS (provavelmente não é conveniente deixar uma número
641 fracionário, mas espero que entenda)
642
643 Analogamente, para a Mopt:
644
645 (600 MC / s) / 2.45 C/I = 244.897959183673 MIPS
646
647 -----
648
649 Questão 2.20
650
651 CPU_Time_MBásica      = (IC * 2.8) / 500 MHz
652 CPU_Time_Mopt         = (IC * 2.45) / 600 MHz
653 speed_up              = ( 2.8 * 600 ) / ( 500 * 2.45 ) ≈ 1.37
654
655 -----
656
657 Questão 2.21
658
659 CPI = (f * t) / IC
660 Supondo que o programa tenha IC instruções na MBásica:
661 quantidade_de_instruções_A = 40% IC
662 quantidade_de_instruções_B = 25% IC
663 quantidade_de_instruções_C = 25% IC
664 quantidade_de_instruções_D = 10% IC
665
666 Na Mcomp as novas relações serão:
667 quantidade_de_instruções_A = 90% 40% IC = 36%      IC
668 quantidade_de_instruções_B = 90% 25% IC = 22.5%    IC
669 quantidade_de_instruções_C = 85% 25% IC = 21.25%   IC
670 quantidade_de_instruções_D = 95% 10% IC = 9.5%      IC
671
672 CPI_médio_novo = (36 * 2 + 22.5 * 3 + 21.25 * 3 + 9.5 * 5) / 89.25 ≈ 2.81
673
674 -----
675
676 Questão 2.22
677
678 CPU_Time_MBásica      = (IC * 2.8) / 500 MHz
679 CPU_Time_Mopt         = (89.25 * IC * 2.81) / (500 MHz * 100)
680 speed_up              = ( 2.8 * 100 ) / ( 89.25 * 2.81 ) ≈ 1.11
681

```

```
680 -----
681
682 Questão 2.23
683
684 CPI_médio_novo = (36 * 2 + 22.5 * 2 + 21.25 * 3 + 9.5 * 4) / 89.25 ≈ 2.451
685 CPU_Time_MBásica = (IC * 2.8) / 500 MHz
686 CPU_Time_Mopt = (89.25 * IC * 2.451) / (600 MHz * 100)
687 speed_up = ( 2.8 * 600 * 100 ) / ( 89.25 * 2.451 * 500 ) ≈ 1.53
```