

In distributed Flynn's SIMD system implementing linear convolution of two n -vectors, then the resulting temporal complexity is equal to:

- a. $188n$
- b. $n\log(n)$
- c. n
- d. n^2
- e. $\log(n)$
- f. $3(n/2)$
- g. $10n^3$



[Clear my choice](#)

For the function $H = a_1a_2a_3 + a_1'a_2 + a_2a_3'$, the number of levels in the corresponding BDT is:

a. 14

b. 12

c. 3

d. 4

e. 2

f. 16

g. 8

h. 9



[Clear my choice](#)

For the function $k = a_1'a_2'a_3' + a_1a_2 + a_2a_3$, the number of leaves in the corresponding non-optimized BDD is:

- a. 5
- b. 2
- c. 14
- d. 12
- e. 8
- f. 15
- g. 16



In distributed Flynn's SIMD system implementing band matrix – band matrix multiplication with array bandwidths $w_1 = 4$ and $w_2 = 4$, then the resulting number of processing elements is equal to:

a. 10

b. 18

c. 7

d. 16

e. 6

f. 8

g. 5

h. 9



For the function $M = b_1b_2 + b_3b_4 + b_5b_6 + b_7b_8$, the number of leaves in the corresponding optimized BDD is:

- a. 6
- b. 7
- c. 3
- d. 4
- e. 11
- f. 2
- g. 9
- h. 16



[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. Knowing that the dot-product computation performs one multiply-add on a single pair of vector elements, so that each floating point operation requires one data fetch, then the limit for the peak speed of this computation (for each one floating point operation) – in terms of [nano seconds, MFLOPS] – is:

a. [100, 8]

b. [200, 5]

c. [100, 2]

d. [400, 6]

e. [200, 4]

f. [200, 8]



[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. Consider introducing a cache of size 32 KB with a latency of 1 ns. We need to multiply two matrices [A] and [B] of dimensions 32×32 . The total number of operations needed to multiply our two matrices is:

- a. 132 M
- b. 48 M
- c. 32 M
- d. 48 K
- e. 32 K
- f. 64 K



[Clear my choice](#)

In distributed Flynn's SIMD system implementing band matrix - vector multiplication with bandwidth w , the temporal complexity of performing the mentioned operation is:

- a. $n + 2w$
- b. $2n + w$
- c. $(n/2)w$
- d. $(n/2) + 2w$
- e. $n^2.w$
- f. $3w + 2n$
- g. $n.w$



[Clear my choice](#)

For the function $k = a_1'a_2'a_3' + a_1a_2 + a_2a_3$, the number of internal nodes in the corresponding BDD is:

a. 6

b. 7

c. 16

d. 11

e. 5

f. 9

g. 12

h. 8



[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. Assume that the used block size is 1 word, then the number of cycles for waiting time before the processor can process the data - for each time a memory request is made - is:



- a. 300 cycles
- b. 200 cycles
- c. 400 cycles
- d. 500 cycles
- e. 100 cycles
- f. 800 cycles

[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. Consider introducing a cache of size 32 KB with a latency of 1 ns. We need to multiply two matrices [A] and [B] of dimensions 32 x 32. The time needed for fetching the two matrices into the cache is:

- a. 100 ns
- b. 200 ns
- c. 400 μ s
- d. 800 μ s
- e. 200 μ s

[Clear my choice](#)

For 16-point FFT, the speedup gain of performing complex multiplications when compared to executing using DFT is:

a. 10.7

b. 16.2

c. 9.4

d. 12.25

e. 7

f. 8.25

g. 17.2



[Clear my choice](#)

For an 8-point distributed FFT implementation, the $O()$ complexity for executing real multiplications is:

a. 27

b. 14

c. 24

d. 48

e. 9

f. 32

g. 12



[Clear my choice](#)

For an 8-point distributed FFT implementation for executing real multiplications, the speedup gain when compared to executing using non-FFT system is:

- a. 8
- b. 11
- c. 16.5
- d. 9.75
- e. 12.5
- f. 13.25
- g. 18.5



[Clear my choice](#)

For 16-point FFT, the speedup gain of performing complex multiplications when compared to executing using DFT is:

- a. 12.25
- b. 10.7
- c. 8.25
- d. 9.4
- e. 16.2
- f. 7
- g. 17.2



[Clear my choice](#)

For the function $M = b_1b_2 + b_3b_4 + b_5b_6 + b_7b_8$, the number of internal nodes in the corresponding optimized BDD with good variable ordering is:

a. 15

b. 9

c. 7

d. 6

e. 16

f. 8

g. 5

h. 14



[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. Consider introducing a cache of size 32 KB with a latency of 1 ns. We need to multiply two matrices [A] and [B] of dimensions 32×32 . The number of words needed to be fetched into the cache is:



- a. 1 K Words
- b. 2 K Words
- c. 1 M Words
- d. 4 K Words
- e. 2 M Words
- f. 4 M Words

[Clear my choice](#)

For an 8-point distributed FFT implementation, the $O()$ complexity for executing real multiplications is:

- a. 14
- b. 27
- c. 32
- d. 24
- e. 9
- f. 12
- g. 48



[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. Consider introducing a cache of size 32 KB with a latency of 1 ns. We need to multiply two matrices [A] and [B] of dimensions 32 x 32. The needed time to perform this matrix multiplication at the rate of 4 instructions / cycle is approximately:

- a. 16 ns
- b. 8 μ s
- c. 4 ns
- d. 4 μ s
- e. 2 ms

[Clear my choice](#)

Consider a processor operating at 2 GHz connected to a DRAM with a latency of $L = 200$ ns initially without using caches. Assume that the processor is of 4-path multi-pipeline architecture and has 2 multiply-add units. The peak processor rating is:

- a. 8 GFLOPS
- b. 2 GFLOPS
- c. 4 GFLOPS
- d. 12 GFLOPS
- e. 6 GFLOPS
- f. 10 GFLOPS

[Clear my choice](#)