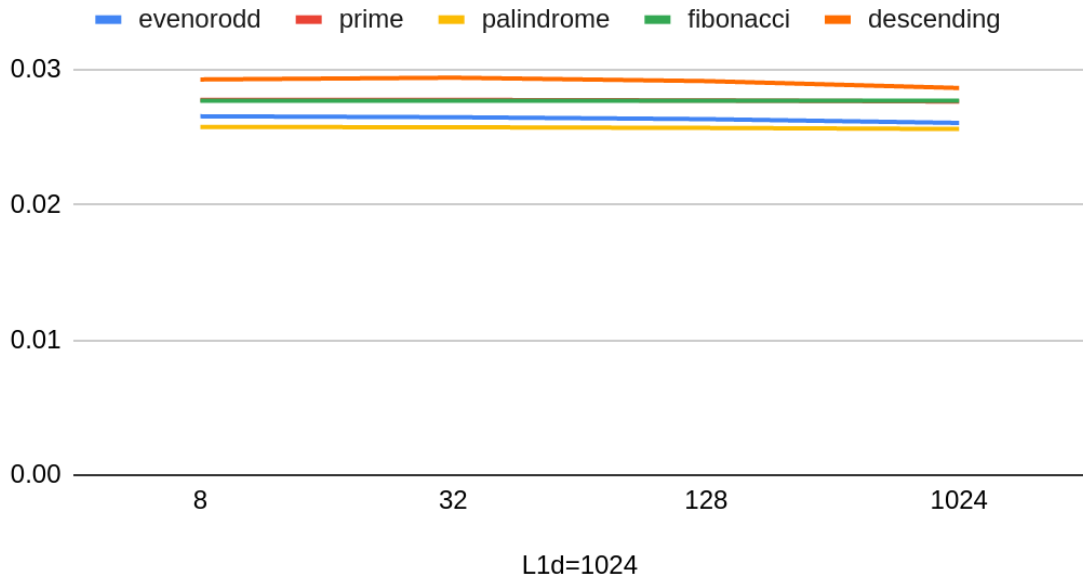


Assignment-6

M V Karthik 200010030 & M Sriram 200020028

Plot - I

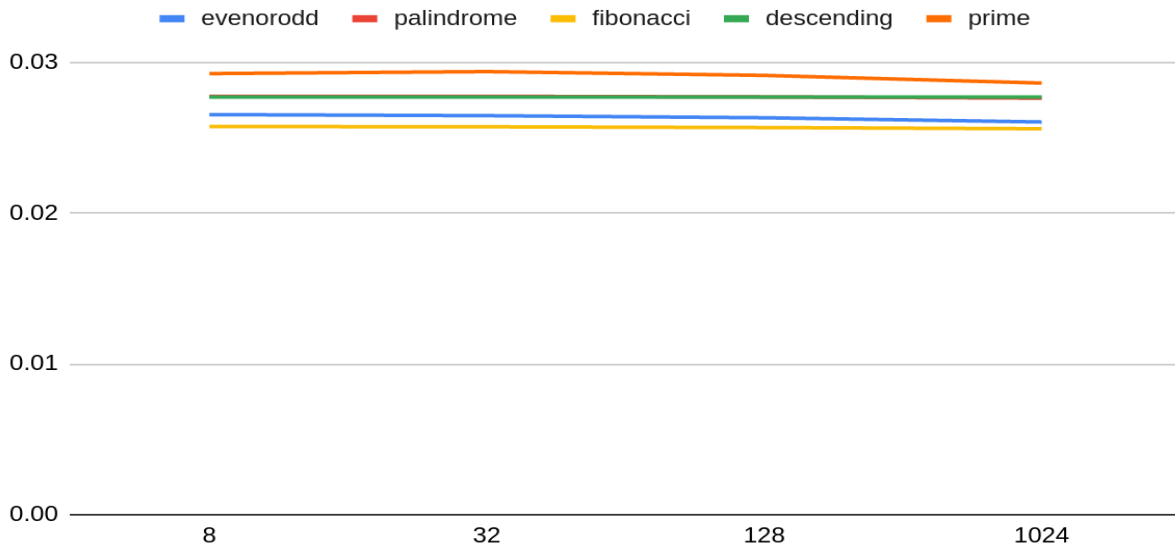
8, 32, 128 and 1024



L1d=1024	8	32	128	1024
evenorodd	0.02653	0.02646	0.02632	0.02604
prime	0.02774	0.02774	0.0277	0.02762
palindrome	0.02574	0.02572	0.02567	0.02559
fibonacci	0.02769	0.02769	0.02769	0.02769
descending	0.02925	0.02938	0.02913	0.02862

Plot - II

8, 32, 128 and 1024



L1i=1024	8	32	128	1024
evenorodd	0.02653	0.02646	0.02632	0.02604
palindrome	0.02774	0.02774	0.0277	0.02762
fibonacci	0.02574	0.02572	0.02567	0.02559
descending	0.02769	0.02769	0.02769	0.02769
prime	0.02925	0.02938	0.02913	0.02862

Q3

As the size of Cache size increases initially, IPC increases because cache is used instead of memory and latency is less than memory, after that we keep increasing cache size, then IPC decreases because of excess latency for high cache.

Q4

```
.data
n:
    10
    .text
main:
    addi %x0, 65535, %x10
    load %x0, $n, %x11
    addi %x0, 0, %x12
    addi %x0, 0, %x13
    addi %x0, 1, %x14
    addi %x12, 1, %x12
    add %x13, %x14, %x8
    store %x13, $n, %x10
    subi %x10, 1, %x10
    beq %x11, %x12, endl
    addi %x12, 1, %x12
    store %x14, $n, %x10
    subi %x10, 1, %x10
    beq %x11, %x12, endl
forloop:
    store %x8, $n, %x10
    subi %x10, 1, %x10
    addi %x12, 1, %x12
    beq %x11, %x12, endl
    addi %x14, 0, %x13
    addi %x8, 0, %x14
    add %x13, %x14, %x8
    jmp forloop
endl:
    end
```

Q5

```
.data
a:
    40
    20
    50
    60
```

```

80
30
10
70
n:
8
.text
main:
    sub %x3, %x3, %x3
    sub %x4, %x4, %x4
    load %x0, $n, %x8
outerloop:
    blt %x3, %x8, innerloop
    end
    addi %x3, 1, %x4
innerloop:
    addi %x3, 1, %x4
innerloopz:
    blt %x4, %x8, swap
    addi %3, 1, %x3
    jmp outerloop
swap:
    load %x3, $a, %x5
    load %x4, $a, %x6
    blt %x5, %x6, exchange
    addi %x4, 1, %x4
    jmp innerloopz
exchange:
    sub %x7, %x7, %x7
    add %x0, %x5, %x7
    store %x6, 0, %x3
    store %x7, 0, %x4
    addi %x4, 1, %x4
    jmp innerloopz

```