# Pipelinening: Report

**Saarang S-112001035**
**Sujit Mandava- 112001043**

**Statistics:**

| Program | Instructions Executed | Clock Cycles Taken | Speed of Processor | Data Hazards* | Control Hazards |
|---|---|---|---|---|---|
| 1-even.asm | 83 | 3386 | 0.02488 | 12 (0) | 760 |
| 2-prime.asm | 7 | 289 | 0.02422 | 5 (0) | 40 |
| 3.descending.asm | 386 | 15624 | 0.02470 | 180 (0) | 2840 |
| 4-histogram.asm | 163 | 6728 | 0.02423 | 204 (0) | 800 |
| 5-fibonnaci.asm | 83 | 3330 | 0.02492 | 6 (0) | 280 |
| 6-arithematic.asm | 54 | 2164 | 0.02495 | 0 (0) | 280 |
| evenorodd.asm | 6 | 250 | 0.024 | 6 (0) | 0 |
| descending.asm | 365 | 14784 | 0.02469 | 180 (0) | 3520 |

**Observations:**

1. The number of cycles taken by the processor increases dramatically from the pipelined processor due to the added latency of the main memory and ALU.
2. One cycle is now equivalent to 40 (or more) cycles due to main memory latency.
3. Thus, the number of data hazards is theoretically 0 as the previous instruction is processed fully before the next instruction is executed, preventing conflicts. The number shows the count of the number of times there is a possible data hazard occurring.
4. More complicated programs, i.e, programs with a higher number of read and write instructions and high branching encounter a higher number of data hazards (RAW hazard) and control hazards
5. Stalling the processor due to mentioned data and control hazards (due to RAW hazards or branching conditions being true) increases the number of cycles taken by the processor.