

Baseline tabela arduino:

```
for (c = 0; c < 1000000; c++) i = j
```

		i = i op 3			i = i op j		
Tipo	Tempo Base	Soma	Or	Mult	Soma	Or	Mult
byte	2462716	2525856	2399568	2652156	2652156	2841596	2841596
int	2715292	2841592	2589004	3031032	3094168	3220468	3599348
float	3599348	12437600	xxxx	10356164	12690176	xxxx	10608752

Cálculo de MIPS (Arduino)

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
byte	15.837821	-15.835814	5.278716	5.278716	2.639358	2.639358
int	7.917656	-7.918409	3.167163	2.639386	1.979508	1.13115001

Cálculo MFloat (Arduino)

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
float	0.11314455	xxxx	0.1479987	0.11000098	xxxx	0.14266548

Cálculo de número de ciclos - CPI (Arduino)

Tipo	Soma	Or	Mult	Soma	Or	Mult
byte	1.01024	-1.0103	3.03104	3.03104	6.06208	6.06208
int	2.0208	-2.0206	5.05184	6.06201	8.08281	14.1448
float	141.4120	xxxx	108.1090	145.4532	xxxx	112.150

Baseline tabela C - Arthur
11th Gen Intel Core i7 / 2803,8 MHz / 16GB RAM

```
for (c = 1; c < 10000000; c++) i = j
```

		i = i op 3			i = i op j		
Tempo	Tempo Base	Soma	Or	Mult	Soma	Or	Mult
char	2.2 ms	17.5 ms	18.7 ms	19.5 ms	20.2 ms	20.4 ms	18 ms
int	2.7 ms	16 ms	16.4 ms	7.9 ms	17.4 ms	17.6 ms	7.8 ms
float	2.8 ms	20.7 ms	xxx	20.3 ms	21 ms	xxx	22 ms

MIPS C - Arthur

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
char	653.594	606.06	578.034	555.555	549.45	632.911
int	751.879	729.927	1923.076	680.272	671.14	1960.784

MFloat C - Arthur

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
float	558.659	xxxx	571.428	549.45	xxxx	520.83

CPI - Arthur

Tipo	Soma	Or	Mult	Soma	Or	Mult
char	5.45688	5.88487	6.1702	6.41986	6.49119	5.63521
int	4.74356	4.88623	1.85463	5.24288	5.31422	1.81896
float	6.38419	xxxx	6.24153	6.49119	xxxx	6.84785

Baseline tabela C - Vitória
MacOS 8 núcleos / 3200 Mhz / 8gb ram

```
for (c = 1; c < 10000000; c++) i = j
```

		i = i op 3			i = i op j		
Tipo	Tempo Base	Soma	Or	Mult	Soma	Or	Mult
char	13ms	24.6ms	24.8ms	31ms	24ms	24.1ms	31.1ms
int	12.3ms	12.3ms	12.3ms	11.8ms	13.1ms	13.2ms	12.9ms
float	12.7ms	38ms	xxxx	37.9ms	37.8ms	xxxx	37.9ms

MIPS C - Vitória

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
char	862.06	847.45	555.55	909.09	909.90	555.48
int	0	0	0	12500	11111.11	16666.66

MFloat C - Vitória

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
float	395.25	xxxx	396.82	398.40	xxx	396.82

CPI - Vitória

Tipo	Soma	Or	Mult	Soma	Or	Mult
char	3.6250	3.6875	5.6250	3.4375	3.4687	5.6562
int	0	0	0	2.5000	2.8125	1.8750
float	7.9062	xxxx	7.8750	7.8437	xxxx	7.8750

Baseline tabela C - Ana
12th Gen Intel(R) Core(TM) i7-1255U
3.200 MHz

```
for (c = 1; c < 10000000; c++) i = j
```

		i = i op 3			i = i op j		
Tempo	Tempo Base	Soma	Or	Mult	Soma	Or	Mult
char	4.2 ms	11.7ms	4.4ms	7.4 ms	6.8 ms	13.2ms	7.8 ms
int	4.1 ms	4.5ms	9.9 ms	6.4 ms	19.4 ms	8.4 ms	9.8 ms
float	4.5 ms	28ms	xxx	35.8 ms	31.5 ms	xxx	33.4 ms

MIPS C - Ana

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
char	1333.33	50000	3125	3846.153	1111.11	2777.77
int	25000	1724.1379	4347.826	653.594	2325.58	1754.38

MFloat C - Ana

	i = i op 3			i = i op j		
Tipo	Soma	Or	Mult	Soma	Or	Mult
float	425.5319	xxx	319.4888	370.3704	xxx	346.0208

CPI - Ana

Tipo	Soma	Or	Mult	Soma	Or	Mult
char	2.3547	361263.3	5.78	4.6964	1.6257	6,5027
int	722526.63	1.0477	4.1545	2.7637	7.767	1.0296

float	4.2448	xxx	5.6538	4.8770	xxx	5.2202
-------	--------	-----	--------	--------	-----	--------

Cálculo SpeedUp Para o Programa em C e Performance

Identificação das máquinas:

M1:

- 11th Gen Intel Core i7
- 2803,8 Mhz
- Windows 11
- GCC

M2:

- 12th Gen Intel(R) Core(TM) i7
- 3.200 Mhz
- Windows 11
- GCC

M3:

- Apple M1 (arm64)
- 3,200 Mhz
- Sonoma 14.0
- Clang

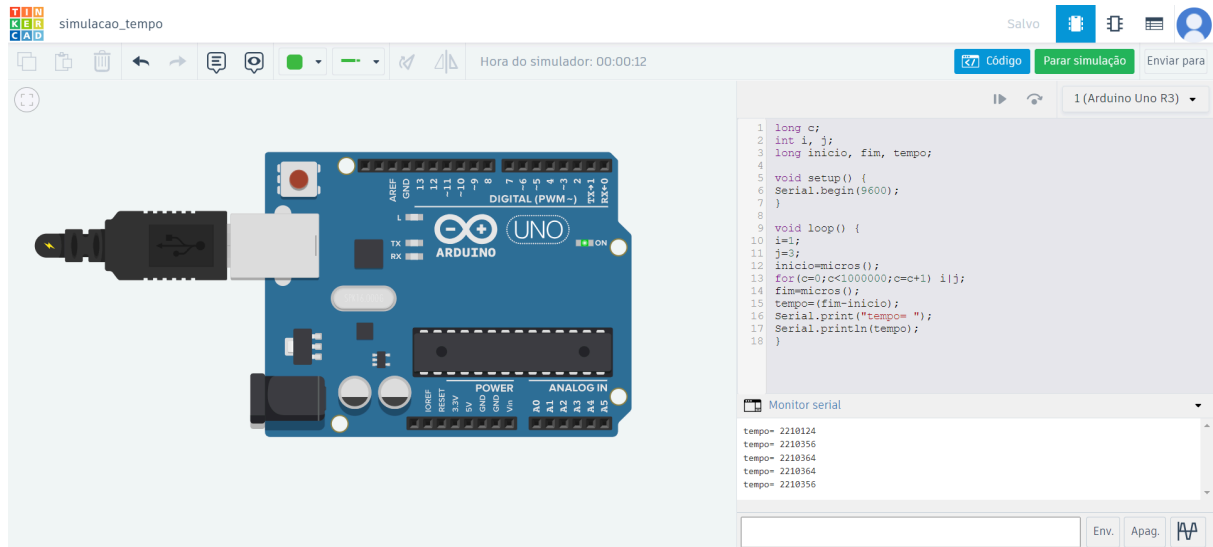
Speed Up inteiros

Programa em C			Performance		
Base	Comparativa	SpeedUp	Base	Comparativa	SpeedUp
M1	M3	1,0992	M1	M2	1,0917
M1	M2	1,9375	M1	M3	1,1625

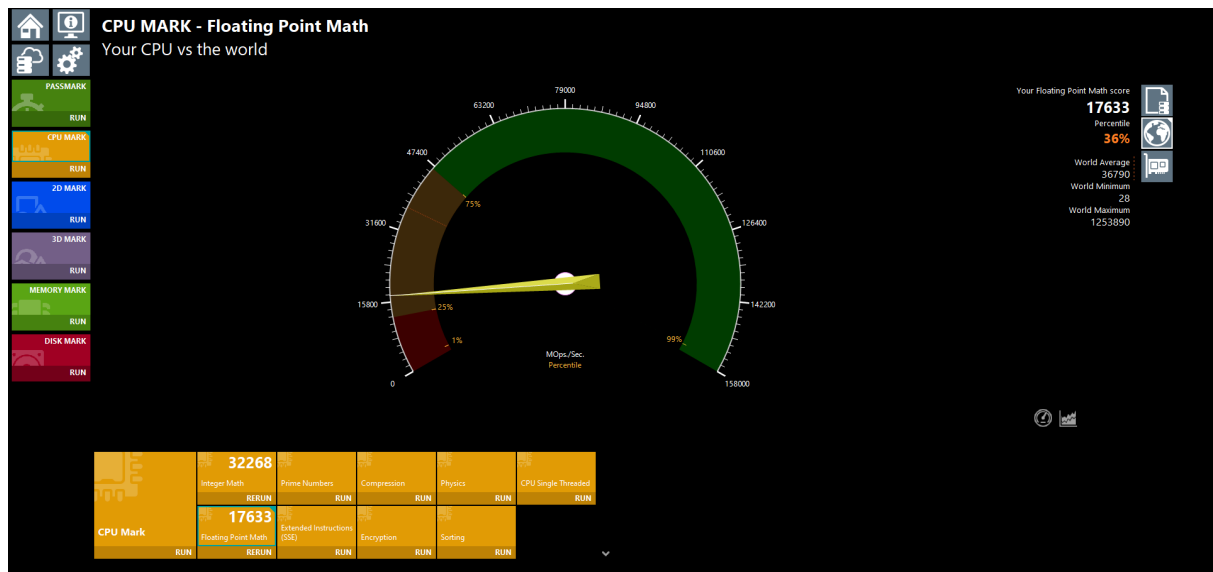
Speed Up Ponto Flutuante

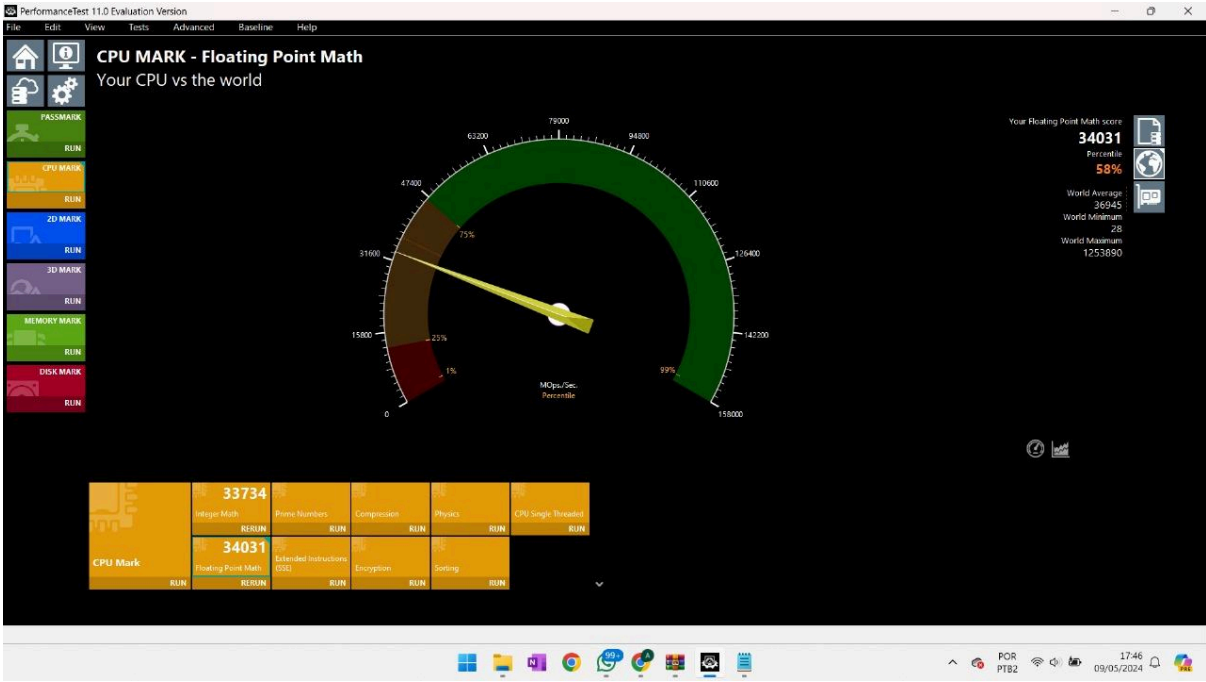
Programa em C			Performance		
Base	Comparativa	SpeedUp	Base	Comparativa	SpeedUp
M3	M1	1,8047	M3	M1	1.8228
M3	M2	1,1779	M3	M2	1.0778

Programa no Arduino



Teste de Performance





PerformanceTest

PerformanceTest CPU Suite

PASSMARK SOFTWARE

System Information

Model: MacBookAir10,1
CPU: Apple M1 (arm64) 8 cores @ 3200 MHz
Memory: 8.0 GiB RAM

Benchmark Website Upload Results

CPU Test Suite

Run	Test	Score
Run	CPU Mark	Incomplete
Run	Integer Math	31677 Million Operations/s
Run	Floating Point Math	36679 Million Operations/s
Run	Prime Numbers	0.0 Million Primes/s
Run	Sorting	0.0 Thousand Strings/s
Run	Encryption	0.0 MB/s
Run	Compression	0.0 KB/s
Run	CPU Single Thread	0.0 Million Operations/s
Run	Physics	0.0 Frames/s
Run	Extended Instructions (NEON)	0.0 Million Matrices/s

Memory Test Suite

Run	Test	Score
Run	Memory Mark	Incomplete
Run	Database Operations	0.0 Thousand Operations/s
Run	Memory Read Cached	0.0 MB/s
Run	Memory Read Uncached	0.0 MB/s
Run	Memory Write	0.0 MB/s
Run	Available RAM	0 Megabytes
Run	Memory Latency	0 Nanoseconds
Run	Memory Threaded	0.0 MB/s

Código em C
Máquina 1

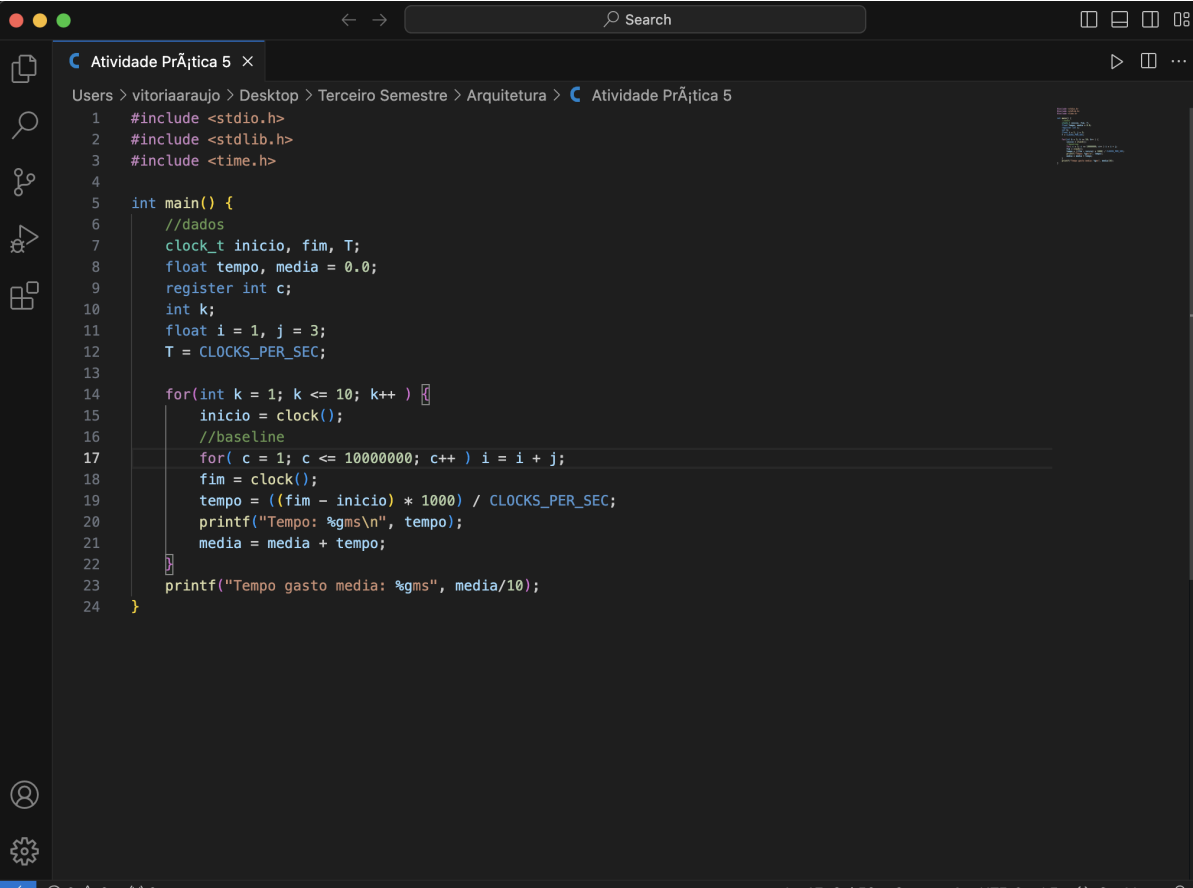
```
main.c 6 X
main.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main()
6  {
7      clock_t inicio, fim, T;
8      float Tempo, media=0;
9      int c;
10     char i='a', j='b', x=3, y=1;
11     int k, num1=1, num2=3;
12     T=CLOCKS_PER_SEC;
13     for (k=1;k<=10;k=k+1)
14     {   inicio=clock();
15         for (c=1;c<=10000000;c=c+1){
16             i=i|3;
17         }
18         fim = clock();
19         Tempo =( (fim - inicio)*1000/CLOCKS_PER_SEC);
20         printf("\nTempo : %g ms.", Tempo);
21         media=media+Tempo;
22     }
23     printf("\nTempo gasto media: %g ms.", media/10);
24 }
```

PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Tempo : 0 ms.
Tempo : 15 ms.
Tempo : 0 ms.
Tempo : 14 ms.
Tempo : 4 ms.
Tempo : 2 ms.
Tempo : 0 ms.
Tempo : 1 ms.
Tempo : 7 ms.
Tempo : 1 ms.
Tempo gasto media: 4.4 ms.
PS C:\Users\anafe\OneDrive\Documentos\CC - PUC\3º período\AB03\TP5\outputs>
```


Programa em C

Máquina 2



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <time.h>
4
5 int main() {
6     //dados
7     clock_t inicio, fim, T;
8     float tempo, media = 0.0;
9     register int c;
10    int k;
11    float i = 1, j = 3;
12    T = CLOCKS_PER_SEC;
13
14    for(int k = 1; k <= 10; k++ ) {
15        inicio = clock();
16        //baseline
17        for( c = 1; c <= 10000000; c++ ) i = i + j;
18        fim = clock();
19        tempo = ((fim - inicio) * 1000) / CLOCKS_PER_SEC;
20        printf("Tempo: %gms\n", tempo);
21        media = media + tempo;
22    }
23    printf("Tempo gasto media: %gms", media/10);
24 }
```

Ln 17, Col 50 Spaces: 4 UTF-8 LF {} C Mac