

# Trabalho Prático 5


Alunos: João Madeira Freitas, João Lucas Curi

## PARTE 1:

Testes de exemplo Arduino:

(Tipo: int) (Operação:  $i = i + j$ ):

```
1  long c;  
2  int i, j;  
3  
4  int controle = 0;  
5  long inicio, fim, tempo;  
6  void setup() {  
7      Serial.begin(9600);  
8  }  
9  
10 void loop() {  
11     i = 1;  
12     j = 3;  
13     inicio = micros();  
14     for(c=0;c<1000000;c=c+1) i=i+j;  
15     fim = micros();  
16     tempo=(fim-inicio);  
17     Serial.print("tempo= ");  
18     Serial.println(tempo);  
19     controle = 1;  
20  
21 }  
22  
23
```

 Monitor serial

tempo= 3094172

(Tipo: byte) (Operação:  $i = i | j$ ):

```
1  long c;  
2  byte i, j;  
3  
4  int controle = 0;  
5  long inicio, fim, tempo;  
6  void setup() {  
7      Serial.begin(9600);  
8  }  
9  
10 void loop() {  
11     i = 1;  
12     j = 3;  
13     inicio = micros();  
14     for(c=0;c<1000000;c=c+1) i=i|j;  
15     fim = micros();  
16     tempo=(fim-inicio);  
17     Serial.print("tempo= ");  
18     Serial.println(tempo);  
19     controle = 1;  
20  
21 }  
22  
23
```



Monitor serial

tempo= 2652156

(Tipo: float) (Operação:  $i = i * j$ ):

```
1  long c;  
2  float i, j;  
3  
4  int controle = 0;  
5  long inicio, fim, tempo;  
6  void setup() {  
7      Serial.begin(9600);  
8  }  
9  
10 void loop() {  
11     i = 1;  
12     j = 3;  
13     inicio = micros();  
14     for(c=0;c<1000000;c=c+1) i=i*j;  
15     fim = micros();  
16     tempo=(fim-inicio);  
17     Serial.print("tempo= ");  
18     Serial.println(tempo);  
19     controle = 1;  
20  
21 }  
22  
23
```



Monitor serial

tempo= 10608752

## Tabelas João Madeira - Parte 1

Tabela 1:

Tipo	Tempo Base	i = i op 3			i = i op j		
		Soma	Or	Mult	Soma	Or	Mult
byte	2462716	2525856	2525856	2652156	2652156	2652156	2841596
int	2715292	2841592	2778444	3031032	3094172	3094172	3599348
float	3220472	12437600	XXXX	10356164	12690176	XXXX	10608752

Tabela 2:

Tipo	MIPS(ATM328P)					
	Constante; i = i op 3			Variável; i = i op j		
	Soma	Or	Multi	Soma	Or	Multi
byte	15.8378	15.8378	5.2787	5.2787	5.2787	2.6394
int	7.9177	15.8348	3.1672	2.6394	2.6394	1.1312
Tipo	MFLOPS(ATM328P)					
	Constante			Variável		
	Soma	Or	Multi	Soma	Or	Multi
float	0.1085	XXXX	0.1401	0.1056	XXXX	0.1353

Tabela 3:

Tipo	CPI					
	Soma	Or	Multi	Soma	Or	Multi
byte	1.010	1.010	3.031	3.031	3.031	6.062
int	2.021	1.010	5.052	6.062	6.062	14.145
float	147.474	XXXX	114.171	151.515	XXXX	118.212

## Tabelas João Lucas - Parte 1

TABELA 1

Tipo	Tempo Base	i = i op 3			i = i op j		
		Soma	Or	Mult	Soma	Or	Mult
byte	24627068	25258532	25258532	26521456	26521456	26521456	28415848
int	27152920	28415844	27784388	30310236	30941700	30941700	35993404
float	32204624	141077692	xxxxxx	103560144	143603540	xxxxxx	106085992

TABELA 2

TIPO	MIPS(ATM328P)					
	Constante			Variável		
	Soma	Or	Multi	Soma	Or	Multi
Byte	1.5836	1.5836	0.5278	0.5278	0.5278	0.2639
Int	0.7918	1.5836	0.3167	0.2639	0.2639	0.1131
TIPO	MFLOPS(ATM328P)					
	Constante			Variável		
	Soma	Or	Multi	Soma	Or	Multi
Float	0.0091	xxxxxxx	0.0140	0.0089	xxxxxxx	0.0135

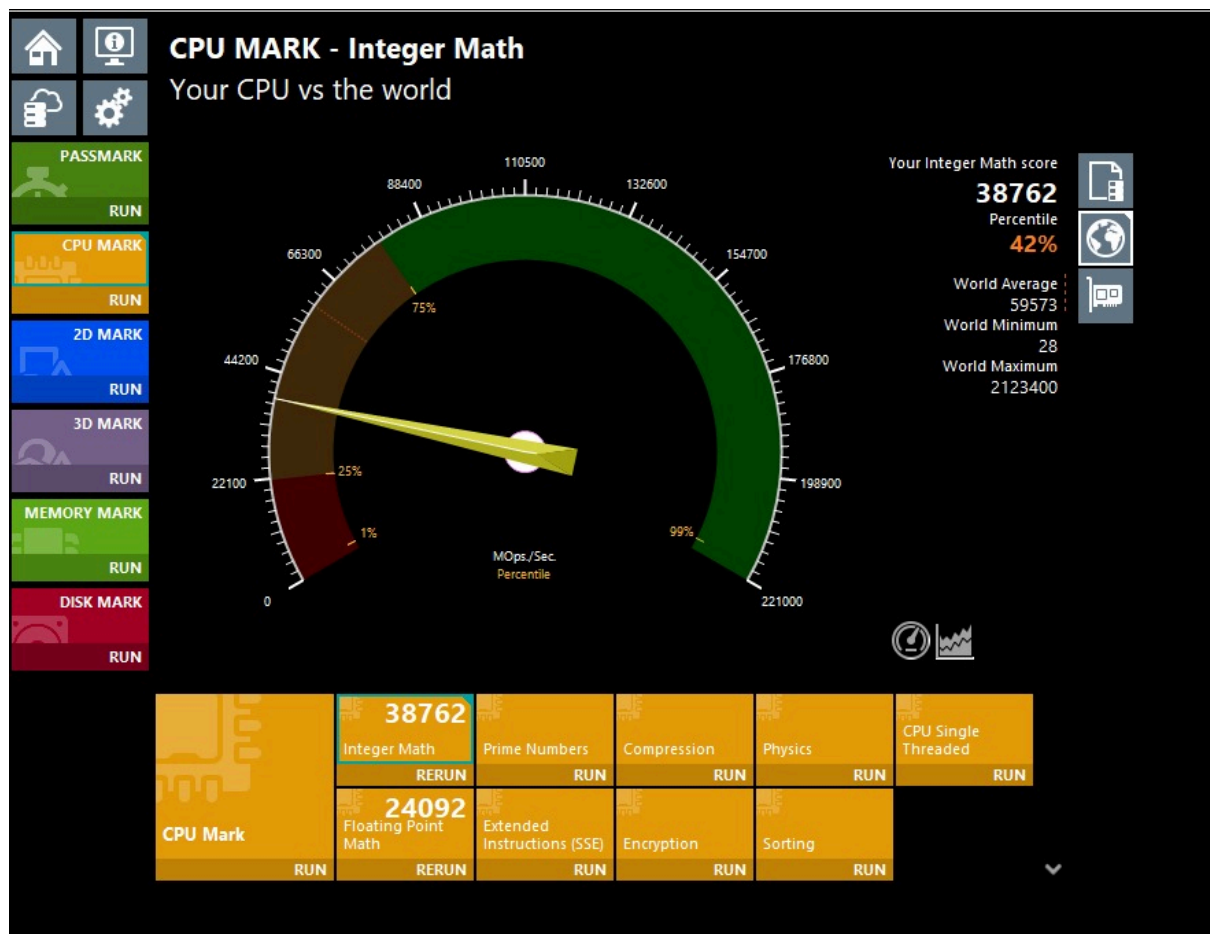
TABELA 3

TIPO	CPI					
	Soma	Or	Multi	Soma	Or	Multi
Byte	10.103	10.103	30.310	30.310	30.310	60.620
Int	20.206	10.103	50.517	60.620	60.620	141.447
Float	1741.969	xxxxx	1141.688	1782.382	xxxxxx	1182.1018

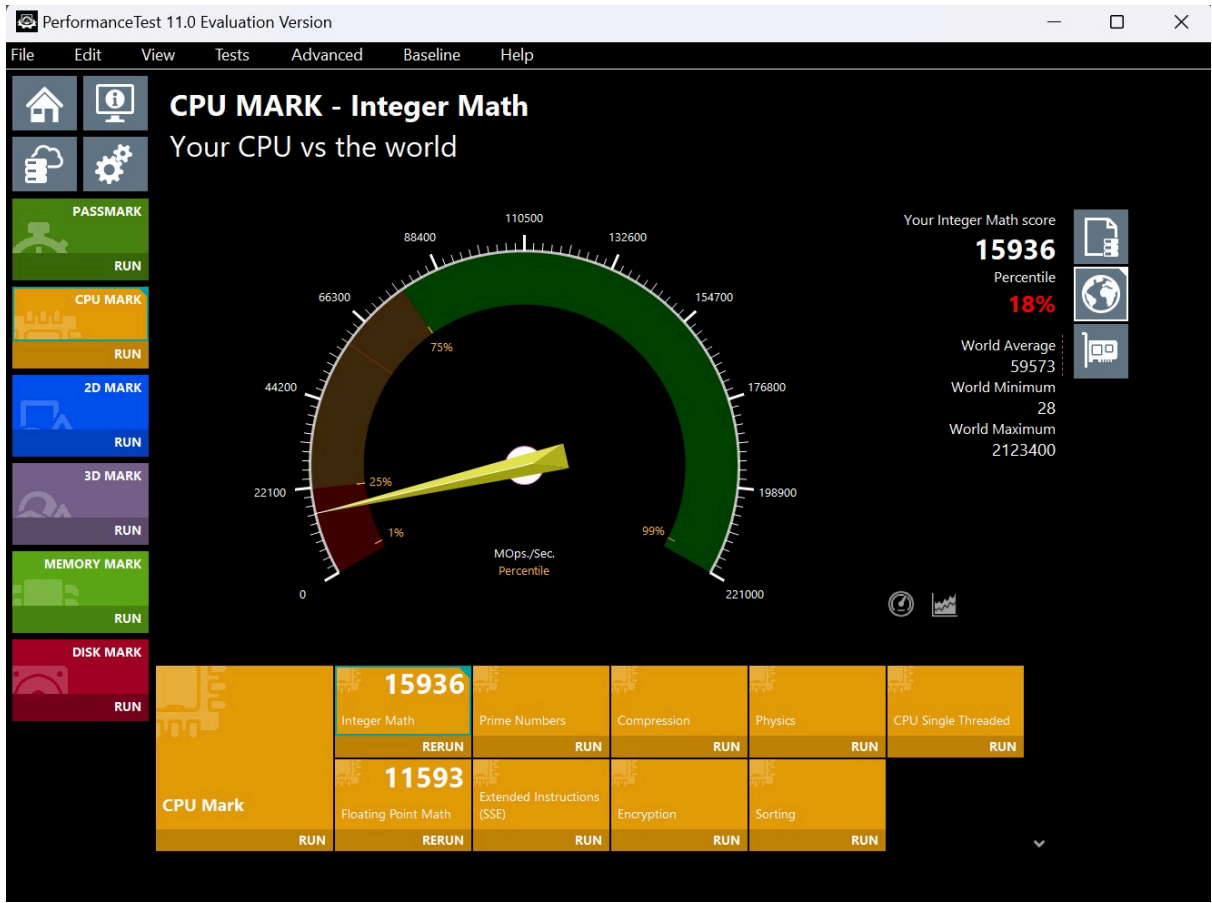
## PARTE 2:

Valores avaliados CPU:

PC João Madeira



PC João Lucas Curi



## Testes de Exemplo (programa em C):

João Madeira:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main() {
6      clock_t inicio, fim, T;
7      float Tempo, media=0;
8      register int c;
9
10     //FLOAT
11     float *i, *j, x=3, y = 1;
12     i = &x;
13     j = &y;
14     // INT
15     int k, num1=1, num2=3;
16
17     //CHAR
18     char *a, *b, c1=1, c2=3;
19     a = &c1;
20     b = &c2;
21
22     T= CLOCKS_PER_SEC;
23     for(k=1;k<=10;k = k+1) {
24         inicio = clock();
25         for(c=1;c<1000000;c=c+1)num1 = num1 + 3;
26         fim = clock();
27         Tempo=((fim-inicio)*1000/CLOCKS_PER_SEC);
28         printf("\nTempo: %g ms.", Tempo);
29         media = media + Tempo;
30     }
31     printf("\nTempo gasto na media: %g ms.", media/10);
32     return 0;
33 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Tempo: 13 ms.  
Tempo: 14 ms.  
Tempo: 13 ms.  
Tempo: 15 ms.  
Tempo: 14 ms.  
Tempo: 13 ms.  
Tempo: 13 ms.  
Tempo: 13 ms.  
Tempo: 13 ms.  
Tempo: 13 ms.  
Tempo: 14 ms.  
Tempo gasto na media: 13.5 ms.  
PS C:\Users\User\OneDrive\Desktop\TestetEX05>

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  int main() {
6      clock_t inicio, fim, T;
7      float Tempo, media=0;
8      register int c;
9
10     //FLOAT
11     float *i, *j, x=3, y = 1;
12     i = &x;
13     j = &y;
14     // INT
15     int k, num1=1, num2=3;
16
17     //CHAR
18     char *a, *b, c1=1, c2=3;
19     a = &c1;
20     b = &c2;
21
22     T= CLOCKS_PER_SEC;
23     for(k=1;k<=10;k = k+1) {
24         inicio = clock();
25         for(c=1;c<1000000;c=c+1)*i = (*i) * (*j);
26         fim = clock();
27         Tempo=((fim-inicio)*1000/CLOCKS_PER_SEC);
28         printf("\nTempo: %g ms.", Tempo);
29         media = media + Tempo;
30     }
31     printf("\nTempo gasto na media: %g ms.", media/10);
32     return 0;
33 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Tempo: 26 ms.  
Tempo: 26 ms.  
Tempo: 27 ms.  
Tempo: 25 ms.  
Tempo: 26 ms.  
Tempo: 26 ms.  
Tempo: 27 ms.  
Tempo: 26 ms.  
Tempo: 26 ms.  
Tempo: 25 ms.  
Tempo gasto na media: 26 ms.  
PS C:\Users\User\OneDrive\Desktop\TestetEX05>



João Lucas Curi:

```
5 int main() {
10 //FLOAT
11 float *i, *j, x=3, y = 1;
12 i = &x;
13 j = &y;
14 // INT
15 int k, num1=1, num2=3;
16
17 //CHAR
18 char *a, *b, c1=1, c2=3;
19 a = &c1;
20 b = &c2;
21
22 T= CLOCKS_PER_SEC;
23 for(k=1;k<=10;k = k+1) {
24     inicio = clock();
25     for(c=1;c<1000000;c=c+1)*a = *a * *b;
26     fim = clock();
27     Tempo=((fim-inicio)*1000/CLOCKS_PER_SEC);
28     printf("\nTempo: %g ms.", Tempo);
29     media = media + Tempo;
```

Tempo: 23 ms.  
Tempo: 22 ms.  
Tempo: 25 ms.  
Tempo: 24 ms.  
Tempo gasto na media: 21.6 ms.

Compiled successfully!

```
5 int main() {
10 //FLOAT
11 float *i, *j, x=3, y = 1;
12 i = &x;
13 j = &y;
14 // INT
15 int k, num1=1, num2=3;
16
17 //CHAR
18 char *a, *b, c1=1, c2=3;
19 a = &c1;
20 b = &c2;
21
22 T= CLOCKS_PER_SEC;
23 for(k=1;k<=10;k = k+1) {
24     inicio = clock();
25     for(c=1;c<1000000;c=c+1)*a = *a * *b;
26     fim = clock();
27     Tempo=((fim-inicio)*1000/CLOCKS_PER_SEC);
28     printf("\nTempo: %g ms.", Tempo);
29     media = media + Tempo;
```

Tempo: 20 ms.  
Tempo: 22 ms.  
Tempo: 20 ms.  
Tempo: 23 ms.  
Tempo gasto na media: 20.2 ms.

Compiled successfully!

## TABELAS João Madeira:

### Parte 2

Tabela 1:

Tipo	Tempo Base	i = i op 3			i = i op j		
		Soma	Or	Multi	Soma	Or	Multi
char	5.7	13.3	13.2	16.1	13.5	14.0	16.7
int	2.9	13.5	13.8	15.7	13.5	13.6	19.7
float	5.3	23.2	XXXX	927.5	23.4	XXXX	26.0

Tabela 2:

Tipo	MIPS(PC João Madeira)					
	Constante			Variável		
	Soma	Or	Multi	Soma	Or	Multi
char	1315.78	1333.33	961.53	1282.05	1204.81	909.09
int	943.39	917.43	781.25	943.39	934.57	595.23
Tipo	MFLOPS(PC João Madeira)					
	Constante			Variável		
	Soma	Or	Multi	Soma	Or	Multi
float	558.65	XXXX	10.84	552.486	XXXX	483.09

Tabela 3:

Tipo	CPI(João Madeira)					
	Soma	Or	Multi	Soma	Or	Multi
char	2.207192	2.178149	3.020368	2.265275	2.410485	3.194619
int	3.078451	3.165577	3.717375	3.078451	3.107494	4.879056
float	5.198517	XXXX	267.82532	5.256601	XXXX	6.011693

## Tabelas João Lucas Curi:

Parte 2

Tabela 1

Tipo	Tempo Base	i = i op 3			i = i op j		
		Soma	Or	Mult	Soma	Or	Mult
char	7.8	18.9	16.5	16.5	20.2	27.5	21.6
int	3.9	14.9	17.6	16.7	15	15.8	29.3
float	6.9	27.3		1121.3	27.8		32.4

Tabela 2

TIPO	MIPS(PC João Lucas)					
	Constante			Variável		
	Soma	Or	Mult	Soma	Or	Mult
CHAR	900.9	1149.4 2	1149.4 2	806.45	507.61	724.63
INT	909.09	729.92	781.25	900.90	840.33	393.70
TIPO	MFLOPS(PC João Lucas)					
	Constante			Variável		
	Soma	Or	Mult	Soma	Or	Mult
FLOAT	490.19	xxxxxx	8.97	478.46	xxxxxx	392.15

TABELA 3|

TIPO	CPI					
	Soma	Or	Mult	Soma	Or	Mult
char	1.990118	1.559823	1.559823	2.223195	3.532013	2.474202
int	1.97219	2.456273	2.294912	1.990119	2.133551	4.553966
float	3.657515	xxxxxxx	199.800775	3.747160	xxxxxxx	4.571895

## Tabelas speedUp

Tabela 4:

Identificação	Programa em C		Performance Test	
	Speed Up (inteiros)	Speed up (FP)	Speed up (inteiros)	Speed up (FP)
PC João Lucas (Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz;  1792.9MHz;  Windows 11 (64bit)  gcc (MinGW GCC-6.3.0-1) 6.3.0	1	1	1	1
PC João Madeira (Intel Core i5 - 10400F @ 2.90GHz;  2904.2MHz;  Windows 11 (64bit)  gcc (MinGW.org GCC-6.3.0-1) 6.3.0)	1.217	1,208	2.439	1.987

- Mesmo SO utilizado
- Mesmo Compilador utilizado

SO e Compilador	Prog em C (inteiros)		Speed Up
	(Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz;  1792.9MHz;	Intel Core i5 - 10400F @ 2.90GHz;  2904.2MHz;	
Windows 11 (64bit)  gcc (MinGW.org GCC-6.3.0-1) 6.3.0)	T(medio) = 18.21	T(medio) = 14.96	1.217