

Capítulo 1:

Apostila de arduino.

Maria Auxiliadora S. Andrade

1. `Serial.print()` transmite dados e imprime na mesma linha e `Serial.println()` recebe códigos a mais e o próximo valor a ser descrito é na linha abaixo

2. Porque são representadas de formas diferentes internamente, pelo tipo da variável e quantidade de bytes.

3.

a) Porque são caracteres de controle, para realizar uma ação.

b. Qual o código correspondente ao 'A'? 65

c. Qual o código correspondente ao 'a'? 97

d. Qual o código correspondente ao '0' (dígito zero)? 48

e. Qual o código correspondente ao ' ' (espaço)? 32

4. Para indicar que é uma variável permanente. Sem o `static` ele vai virar uma variável local e o valor não vai permanecer, vai ficar se destruindo toda as vezes que o loop reinicia.

5. Porque na tabela ASCII só vai até o 255.

6. Quando chega ao 255 ele continua em ordem crescente, como se tivesse voltado ao 0

7.

8.

a) Qual o código correspondente ao '{'? 123

b) Quais os códigos correspondentes ao 'a'? e ao 'á'? e ao 'à'? a: 97 , á: 225 e à: 224

9. Na comunicação serial a informação é enviado uma após a outra, como cada letra representa uma informação no programa, cada caractere será entregue uma após a outra

10) Porque não tem necessidade de manter seus valores

11.

a) Dígito numérico";

```
void setup()
```

```
{ Serial.begin(9600);
```

```
}
```

```

void loop() {
  delay (10);
  byte n = 0;
  char c;
  if ( Serial.available() > 0 ) {
    c = Serial.read();
    n = c;
    Serial.print ( n );
    Serial.print ( " | " );
    Serial.println ( c );
    if(n >= 0&&n <= 9){
      Serial.println("Categoria: Dígito um número");
    }
  }
}

```

```

b)
void setup() {
  Serial.begin(9600);
}
void loop() {
  delay (10);
  byte n = 0;
  char c;
  if ( Serial.available() > 0 ) {
    c = Serial.read();
    n = c;
    Serial.print ( n );
    Serial.print ( " | " );
    Serial.println ( c );
    if(n >= 97&&n <= 122){
      Serial.println("Categoria: Letra maiúscula");
    }
  }
}

```

```

c)
void setup() {
  Serial.begin(9600);
}
void loop() { delay (10);
  byte n = 0;
  char c;
  if ( Serial.available() > 0 ) {
    c = Serial.read();
    n = c;
    Serial.print ( n );
    Serial.print ( " | " );
    Serial.println ( c );
    if(n >= 97&&n <= 122){
      Serial.println("Categoria: Letra minúscula");
    }
  }
}

```

```
}  
}  
}
```

```
d)  
void setup() {  
  Serial.begin(9600);  
}  
void loop() {  
  delay (10);  
  byte n = 0;  
  char c;  
  if ( Serial.available() > 0 ) {  
    c = Serial.read();  
    n = c;  
    Serial.print ( n );  
    Serial.print ( " | " );  
    Serial.println ( c );  
    if(n != 0&& n != 122){  
      Serial.println("Categoria: Outros");  
    }  
  }  
}
```

```
12.void setup() {  
  Serial.begin(9600);  
}  
void loop() {  
  delay (10);  
  byte n = 0;  
  char c;  
  if ( Serial.available() > 0 ) {  
    c = Serial.read();  
    n = c;  
    Serial.print ( n );  
    Serial.print ( " | " );  
    Serial.println ( c );  
    if(n <= 127){  
      else( n=0());  
    }  
  }  
}
```

Capítulo 2:

1.As globais podem ser acessadas de qualquer função, enquanto a local apenas dentro da função que ela foi inserida. Local : no inicio de um subalgoritmo e global: no inicio do algoritmo.

2.a)

```
const byte led1    = 13;
const byte botao1 =  5;
```

```
void setup() {
    pinMode ( led1, OUTPUT );
    pinMode ( botao1, INPUT );
    Serial.begin( 9600 );
}

void loop() {
    delay (10);
    bool bt = digitalRead(botao1);
    if(digitalRead(botao1) == LOW){
        digitalWrite(led1, LOW);
    }else {
        digitalWrite(led1, HIGH);
    }
}
```

b) const byte led1 = 13;
const byte botao1 = 5;

```
void setup() {
    pinMode ( led1, OUTPUT );
    pinMode ( botao1, INPUT );
    Serial.begin( 9600 );
}

void loop() {
    delay (10);
    bool bt = digitalRead(botao1);
    if(digitalRead(botao1) == LOW){
        digitalWrite(led1, HIGH);
    }else {
        digitalWrite(led1, LOW);
    }
}
```

3.a)

```
const byte led1    = 13;
const byte botao1 =  5;
const byte botao2 =  4;
```

```
void setup() {
    pinMode ( led1, OUTPUT );
```

```

    pinMode ( botao1, INPUT );
    Serial.begin( 9600 );
}

void loop() {
    delay (10);
    bool bt = digitalRead(botao1);
    if(digitalRead(botao1) == HIGH){
        digitalWrite(led1, HIGH);
    }
    if(digitalRead(botao2) == HIGH){
        digitalWrite(led1, LOW);
    }
}

b) const byte led1    = 13;
const byte botao1 = 5;
const byte botao2 = 4;

void setup() {
    pinMode ( led1, OUTPUT );
    pinMode ( botao1, INPUT );
    Serial.begin( 9600 );
}

void loop() {
    delay (10);
    bool bt = digitalRead(botao1);
    if(digitalRead(botao1) == HIGH && digitalRead(botao2) == HIGH){
        digitalWrite(led1, HIGH);
    }
    if(digitalRead(botao1) == LOW && digitalRead(botao2) == LOW){
        digitalWrite(led1, LOW);
    }
}

c) const byte led1    = 13;
const byte botao1 = 5;
const byte botao2 = 4;

void setup() {
    pinMode ( led1, OUTPUT );
    pinMode ( botao1, INPUT );
    Serial.begin( 9600 );
}

void loop() {
    delay (10);
    bool bt = digitalRead(botao1);
    if(digitalRead(botao1) == HIGH && digitalRead(botao2) == HIGH){
        digitalWrite(led1, HIGH);
    }
    if(digitalRead(botao1) == LOW || digitalRead(botao2) == LOW){
        digitalWrite(led1, LOW);
    }
}

4. const byte LED1    = 13;
const byte BOTA01 = 5;
char c;

```

```

void setup() {
  pinMode ( LED1, OUTPUT );
  pinMode ( BOTA01, INPUT );
  Serial.begin( 9600 );
}

void loop() {
  delay (10);
  if(Serial.available() > 0){
    c = Serial.read();
  }
  if(c == 'A'){
    digitalWrite(LED1, HIGH);
  }
  if(c == 'a'){
    digitalWrite(LED1, LOW);
  }
}
}

```

5. const byte LED1 = 13;
 const byte BOTA01 = 5;
 char c;

```

void setup() {
  pinMode ( LED1, OUTPUT );
  pinMode ( BOTA01, INPUT );
  Serial.begin( 9600 );
}

void loop() {
  delay (10);
  if(Serial.available() > 0){
    c = Serial.read();
  }
  if(c >= '65' && c <= '96'){
    digitalWrite(LED1, HIGH);
  }
  if(c <= '97' && c <= '122'){
    digitalWrite(LED1, LOW);
  }
}

```

6.1023 e 0, respectivamente

7.

```

const byte POT = A0;
int led13 = 13;
int led13 = 13;

void setup() {
  Serial.begin(9600);
  pinMode(led13, OUTPUT);
}

void loop() {
  delay (10);
  int valor_pot = analogRead(POT);

```

```

Serial.println( valor_pot );
if (valor_pot > 511){
digitalWrite(led13, HIGH);
}else{ digitalWrite(led13, LOW);
}
}

```

8.

```

const byte POT = A0;
int led13 = 13;
int led12 = 12;
int led11 = 11;

void setup() {
Serial.begin(9600);
pinMode(led13, OUTPUT);
pinMode(led12, OUTPUT);
pinMode(led11, OUTPUT);
}
void loop() {
delay (10);
int valor_pot = analogRead(POT);
Serial.println( valor_pot );
if (valor_pot > 0 && valor_pot < 342){
digitalWrite(led13, HIGH);
} else { digitalWrite(led13, LOW);
}
if (valor_pot > 342 && valor_pot < 683){
digitalWrite(led12, HIGH);
} else { digitalWrite(led12, LOW);
} if (valor_pot > 683 ){ digitalWrite(led11, HIGH);
} else { digitalWrite(led11, LOW); ]
}
}

```

9.

```

const byte POT = A0;
const byte LDR = A1;
int led13 = 13;

void setup() {
Serial.begin(9600);
pinMode(led13, OUTPUT);
}

void loop() {
delay (10);
int valor_pot = analogRead(POT);
int valor_ldr = analogRead(LDR);
Serial.println( valor_pot );
if (valor_pot > valor_ldr){
digitalWrite(led13, HIGH);
} else { digitalWrite(led13, LOW);
}
}
}

```

10.

a)

25°C, valor correspondente de 153 no monitor serial.

58°C, valor correspondente de 217 no monitor serial.

código:

```
const byte TMP = A2;
void setup() { Serial.begin(9600);
}
void loop() {
  delay (10);
  int valor_tmp = analogRead(TMP);
  Serial.println( valor_tmp );
}
```

b)

```
const byte TMP = A2;
void setup() {
  Serial.begin(9600);
}
```

```
void loop() {
  delay (10);
  int valor_tmp = analogRead(TMP);
  int ctmp = map(valor_tmp, 153, 217, 25, 58);
  Serial.println( ctmp );
}
```

c)

```
const byte TMP = A2;
int led13 = 13;
void setup() {
  Serial.begin(9600);
}
```

```
void loop() {
  delay (10);
  int valor_tmp = analogRead(TMP);
  int ctmp = map(valor_tmp, 153, 217, 25, 58);
  Serial.println( ctmp );
  if (analogRead(TMP) > 60){
    digitalWrite(led13, HIGH);
  } else { digitalWrite(led13, LOW);
  }
}
```