

nächstes mal in der Abgabe
die richtigen Matrikelnummern



Alle Aufgaben:

1.1)

```
const unsigned int led = 13;
const unsigned int t_on=1000, t_off=1000;
```

```
void setup () {
  pinMode ( led , OUTPUT );
}
void loop () {
  digitalWrite ( led , HIGH );
  delay (t_on);
  digitalWrite ( led , LOW );
  delay(t_off);
}
```

✓ 0,5 P

1.2.a)

```
const unsigned int led = 13;
const unsigned int t_on=2500, t_off=2500;
```

```
void setup () {
  pinMode ( led , OUTPUT );
}
void loop () {
  digitalWrite ( led , HIGH );
  delay (t_on);
  digitalWrite ( led , LOW );
  delay(t_off);
}
```

✓ 0,5 P

1.2.b)

```
const unsigned int led = 13;
const unsigned int t_on=3000, t_off=1000;
```

```
void setup () {
  pinMode ( led , OUTPUT );
}
void loop () {
  digitalWrite ( led , HIGH );
  delay (t_on);
  digitalWrite ( led , LOW );
  delay(t_off);
}
```

0,5 P

1.2.c)

```
const unsigned int led = 13;
```

```
const unsigned int t_on=500, t_off=500;
```

```
void setup () {  
  pinMode ( led , OUTPUT );  
}  
void loop () {  
  digitalWrite ( led , HIGH );  
  delay (t_on);  
  digitalWrite ( led , LOW );  
  delay(t_off);  
}
```

0.5P

2.1)

```
int led = 13;  // led to use for output (13 is built-in)
```

```
int dit_delay = 500;           // length of a dit in ms
```

```
int dah_delay = dit_delay * 3; // length of a dah in ms
```

```
// dit() and dat() already include dit_delay
```

```
int letter_delay = dah_delay - dit_delay;
```

```
// letters already include letter delay
```

```
int word_delay  = dit_delay * 7 - letter_delay;
```

```
void dit() {  
  Serial.print(".");  
  
  // send a dit  
  digitalWrite(led, HIGH);  
  delay(dit_delay);  
  
  digitalWrite(led, LOW);  
  delay(dit_delay);  
}
```

```
void dah() {  
  Serial.print("-");  
  
  // send a dah  
  digitalWrite(led, HIGH);  
  delay(dah_delay);  
  
  digitalWrite(led, LOW);  
  delay(dit_delay);  
}
```

```

void pause_letter() {
    Serial.println("");
    delay(letter_delay);
}

void pause_word() {
    Serial.println("");
    delay(word_delay);
}

void morse_S() {
    dit(); dit(); dit();
    pause_letter();
}

void morse_O() {
    dah(); dah(); dah();
    pause_letter();
}

void morse_SOS() {
    morse_S(); morse_O(); morse_S();
    pause_word();
}

// send n in morse code
void morse_digit(int n) {
    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...");

    if (n == 0) {
        dah(); dah(); dah(); dah(); dah();
    }
    if (n == 1) {
        dit(); dah(); dah(); dah(); dah();
    }
    if (n == 2) {
        dit(); dit(); dah(); dah(); dah();
    }
    if (n == 3) {
        dit(); dit(); dit(); dah(); dah();
    }
    if (n == 4) {
        dit(); dit(); dit(); dit(); dah();
    }
}

```

```

    if (n == 5) {
        dit(); dit(); dit(); dit(); dit();
    }
    if (n == 6) {
        dah(); dit(); dit(); dit(); dit();
    }
    if (n == 7) {
        dah(); dah(); dit(); dit(); dit();
    }
    if (n == 8) {
        dah(); dah(); dah(); dit(); dit();
    }
    if (n == 9) {
        dah(); dah(); dah(); dah(); dit();
    }
    pause_letter();

    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...done");
}

void morse_number(int n) {
    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...");

    if (n >= 10) {
        morse_number(n / 10);
    }
    morse_digit(n % 10);


    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...done");
}

void setup() {
    // put your setup code here, to run once:


    // Configure LED as output
    pinMode(led, OUTPUT);

    // Use serial port as output
    Serial.begin(9600);
}

```



```
void morse_letter(char c) {
  Serial.print("morse_letter(");
  Serial.print(c);
  Serial.println(")...");
  switch (c) {
    case 'a':{dit();dah();break;}
    case 'b':{dah();dit();dit();dit();break;}
    case 'c':{dah();dit();dah();dit();break;}
    case 'd':{dah();dit();dit();break;}
    case 'e':{dit();break;}
    case 'f':{dit();dit();dah();dit();break;}
    case 'g':{dah();dah();dit();break;}
    case 'h':{dit();dit();dit();dit();break;}
    case 'i':{dit();dit();break;}
    case 'j':{dit();dah();dah();dah();break;}
    case 'k':{dah();dit();dah();break;}
    case 'l':{dit();dah();dit();dit();break;}
    case 'm':{dah();dah();break;}
    case 'n':{dah();dit();break;}
    case 'o':{dah();dah();dah();break;}
    case 'p':{dit();dah();dah();dit();break;}
    case 'q':{dah();dah();dit();dah();break;}
    case 'r':{dit();dah();dit();break;}
    case 's':{dit();dit();dit();break;}
    case 't':{dah();break;}
    case 'u':{dit();dit();dah();break;}
    case 'v':{dit();dit();dit();dah();break;}
    case 'w':{dit();dah();dah();break;}
    case 'x':{dah();dit();dit();dah();break;}
    case 'y':{dah();dit();dah();dah();break;}
    case 'z':{dah();dah();dit();dit();break;}
  }
  pause_letter();
  Serial.print("morse_letter(");
  Serial.print(c);
  Serial.println(")...done");
}
```



```
void morse_word(String s){
  Serial.print("morse_word(");
  Serial.print(s);
  Serial.println(")...");
  for (int i = 0; i < s.length(); i++) {
    morse_letter(s[i]);
  }
}
```

```

    pause_word();
    Serial.print("morse_word(");
    Serial.print(s);
    Serial.println(")...done");
}

```

```

void loop() {
    // put your main code here, to run repeatedly:
    morse SOS();
    morse_number(5024);
    morse_word("saarland");
    morse_word("university");
}

```

✓

3/3

```

2.2)
int led1 = 12; // led to use for output (13 is built-in)
int led2 = 13;
int dit_delay = 500; // length of a dit in ms
int dah_delay = dit_delay * 3; // length of a dah in ms

```

```

// dit() and dah() already include dit_delay
int letter_delay = dah_delay - dit_delay;

```

```

// letters already include letter delay
int word_delay = dit_delay * 7 - letter_delay;

```

✓

```

void dit(int a) {
    Serial.print(".");

    // send a dit
    digitalWrite(a, HIGH);
    delay(dit_delay);

    digitalWrite(a, LOW);
    delay(dit_delay);
}

```

✓

```

void dah(int a) {
    Serial.print("-");

    // send a dah
    digitalWrite(a, HIGH);
    delay(dah_delay);

    digitalWrite(a, LOW);
    delay(dit_delay);
}

```

```

}

void pause_letter() {
    Serial.println("");
    delay(letter_delay);
}

void pause_word() {
    Serial.println("");
    delay(word_delay);
}

void morse_S() {
    dit(led1); dit(led1); dit(led1);
    pause_letter();
}

void morse_O() {
    dah(led1); dah(led1); dah(led1);
    pause_letter();
}

void morse_SOS() {
    morse_S(); morse_O(); morse_S();
    pause_word();
}

// send n in morse code
void morse_digit(int n) {
    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...");

    if (n == 0) {
        dah(led1); dah(led1); dah(led1); dah(led1); dah(led1);
    }
    if (n == 1) {
        dit(led1); dah(led1); dah(led1); dah(led1); dah(led1);
    }
    if (n == 2) {
        dit(led1); dit(led1); dah(led1); dah(led1); dah(led1);
    }
    if (n == 3) {
        dit(led1); dit(led1); dit(led1); dah(led1); dah(led1);
    }
    if (n == 4) {

```

```

        dit(led1); dit(led1); dit(led1); dit(led1); dah(led1);
    }
    if (n == 5) {
        dit(led1); dit(led1); dit(led1); dit(led1); dit(led1);
    }
    if (n == 6) {
        dah(led1); dit(led1); dit(led1); dit(led1); dit(led1);
    }
    if (n == 7) {
        dah(led1); dah(led1); dit(led1); dit(led1); dit(led1);
    }
    if (n == 8) {
        dah(led1); dah(led1); dah(led1); dit(led1); dit(led1);
    }
    if (n == 9) {
        dah(led1); dah(led1); dah(led1); dah(led1); dit(led1);
    }
    pause_letter();

    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...done");
}

void morse_number(int n) {
    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...");

    if (n >= 10) {
        morse_number(n / 10);
    }
    morse_digit(n % 10);

    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...done");
}

void setup() {
    // put your setup code here, to run once:

    // Configure LED as output
    pinMode(led1, OUTPUT); ✓
    pinMode(led2, OUTPUT); ✓
    // Use serial port as output

```



```

    Serial.begin(9600);
}
    ✓
void morse_letter(char c, int a) {
    Serial.print("morse_letter(");
    Serial.print(c);
    Serial.println(")...");
    switch (c) {
        case 'b':{dah(a);dit(a);dit(a);dit(a);break;}
        case 'a':{dit(a);dah(a);break;}
        case 'c':{dah(a);dit(a);dah(a);dit(a);break;}
        case 'd':{dah(a);dit(a);dit(a);break;}
        case 'e':{dit(a);break;}
        case 'f':{dit(a);dit(a);dah(a);dit(a);break;}
        case 'g':{dah(a);dah(a);dit(a);break;}
        case 'h':{dit(a);dit(a);dit(a);dit(a);break;}
        case 'i':{dit(a);dit(a);break;}
        case 'j':{dit(a);dah(a);dah(a);dah(a);break;}
        case 'k':{dah(a);dit(a);dah(a);break;}
        case 'l':{dit(a);dah(a);dit(a);dit(a);break;}
        case 'm':{dah(a);dah(a);break;}
        case 'n':{dah(a);dit(a);break;}
        case 'o':{dah(a);dah(a);dah(a);break;}
        case 'p':{dit(a);dah(a);dah(a);dit(a);break;}
        case 'q':{dah(a);dah(a);dit(a);dah(a);break;}
        case 'r':{dit(a);dah(a);dit(a);break;}
        case 's':{dit(a);dit(a);dit(a);break;}
        case 't':{dah(a);break;}
        case 'u':{dit(a);dit(a);dah(a);break;}
        case 'v':{dit(a);dit(a);dit(a);dah(a);break;}
        case 'w':{dit(a);dah(a);dah(a);break;}
        case 'x':{dah(a);dit(a);dit(a);dah(a);break;}
        case 'y':{dah(a);dit(a);dah(a);dah(a);break;}
        case 'z':{dah(a);dah(a);dit(a);dit(a);break;}
    }
    pause_letter();
    Serial.print("morse_letter(");
    Serial.print(c);
    Serial.println(")...done");
}
    ✓

```

```

void morse_word(String s, int a){
    Serial.print("morse_word(");
    Serial.print(s);
    Serial.println(")...");
    for (int i = 0; i < s.length(); i++) {

```

```

    morse_letter(s[i], a);
}
pause_word();
Serial.print("morse_word(");
Serial.print(s);
Serial.println(")...done");
}

void loop() {
    // put your main code here, to run repeatedly:
    morse_SOS();
    morse_number(5024);
    morse_word("saarland", led1);
    morse_word("university", led2);
}

```

✓
+ 1 Bonuspunkt

2.3

```
int led = 13; // led to use for output (13 is built-in)
```

```
int dit_delay = 500; // length of a dit in ms
int dah_delay = dit_delay * 3; // length of a dah in ms
```

```
// dit() and dat() already include dit_delay
int letter_delay = dah_delay - dit_delay;
```

```
// letters already include letter delay
int word_delay = dit_delay * 7 - letter_delay;
```

```
void dit() {
    Serial.print(".");

    // send a dit
    digitalWrite(led, HIGH);
    delay(dit_delay);

    digitalWrite(led, LOW);
    delay(dit_delay);
}

```

```
void dah() {
    Serial.print("-");

    // send a dah
    digitalWrite(led, HIGH);
    delay(dah_delay);
}

```

```

        digitalWrite(led, LOW);
        delay(dit_delay);
    }

void pause_letter() {
    Serial.println("");
    delay(letter_delay);
}

void pause_word() {
    Serial.println("");
    delay(word_delay);
}

void morse_S() {
    dit(); dit(); dit();
    pause_letter();
}

void morse_O() {
    dah(); dah(); dah();
    pause_letter();
}

void morse_SOS() {
    morse_S(); morse_O(); morse_S();
    pause_word();
}

// send n in morse code
void morse_digit(int n) {
    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...");

    if (n == 0) {
        dah(); dah(); dah(); dah(); dah();
    }
    if (n == 1) {
        dit(); dah(); dah(); dah(); dah();
    }
    if (n == 2) {
        dit(); dit(); dah(); dah(); dah();
    }
    if (n == 3) {

```

```

        dit(); dit(); dit(); dah(); dah();
    }
    if (n == 4) {
        dit(); dit(); dit(); dit(); dah();
    }
    if (n == 5) {
        dit(); dit(); dit(); dit(); dit();
    }
    if (n == 6) {
        dah(); dit(); dit(); dit(); dit();
    }
    if (n == 7) {
        dah(); dah(); dit(); dit(); dit();
    }
    if (n == 8) {
        dah(); dah(); dah(); dit(); dit();
    }
    if (n == 9) {
        dah(); dah(); dah(); dah(); dit();
    }
    pause_letter();

    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...done");
}

```

```

void morse_number(int n) {
    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...");

    if (n >= 10) {
        morse_number(n / 10);
    }
    morse_digit_even(n % 10); //changed

    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...done");
}

```

```

void morse_digit_even(int n) {
    if (n%2==0) {
        morse_digit(n);
    }
}

```

✓

✓

```

}

void setup() {
  // put your setup code here, to run once:

  // Configure LED as output
  pinMode(led, OUTPUT);

  // Use serial port as output
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  morse_SOS();
  morse_number(876543210);
}

```

2/2

3.a)

Fehlermeldungen passiert beim Übersetzen dieses Programms, weil es viele Syntax-Fehler gibt. Z.b die setup funktion hat keine "()". Das Programms hat auch eine Logikfehler in der blink_if_even Funktion.

3.b)

Es gibt Fehlermeldungen und diese bedeutet dass es Syntax-Fehlen gibt.

3.c)

Es gibt Fehlermeldungen.

3.d)

```

const int led_red = 12;
const int led_green = 12+1;

int counter = 1;
void blink_if_even (int n ) {
  if ( n % 2 == 0 ) { // even
    digitalWrite ( led_red , HIGH ) ;
    digitalWrite ( led_green , LOW ) ;
  } else { // odd
    digitalWrite ( led_red , LOW ) ;
    digitalWrite ( led_green , HIGH ) ;
  }
}

```

```

void setup() {
  pinMode ( led_red , OUTPUT );
  pinMode ( led_green , OUTPUT );
}
void loop() {
  blink_if_even ( counter );
  counter = counter + 1;
  delay (1000);
}

```

✓

✓

5/5

3 bonus)

Mit einer einfacheren Methode um die LED abwechselnd blinken zu lassen.

Man kann im loop die erste LED eine sekunde aufleuchte lassen und danach die zweite, sodass der loop dann diese abfolge wiederholt.

Wenn man in einem loop durchlauf nur einmal die LEDs ändern möchte, kann man die methode blink_if_even durch eine swap methode ersetzen,

und den counter braucht man auch nicht:

```

int led1=12;
int led2=13;
int tausch=0;

```

```

void setup() {
  pinMode ( led2 , OUTPUT );
  pinMode ( led2 , OUTPUT );
}
void swap(){
  digitalWrite(led1, HIGH);
  digitalWrite(led2, LOW);
  tausch=led1;
  led1=led2;
  led2=tausch;
}

```

```

void loop(){
  swap();
  delay(1000);
}

```

✓ + 2 Bonuspunkte

12/12 + 3 Bonuspunkte

15