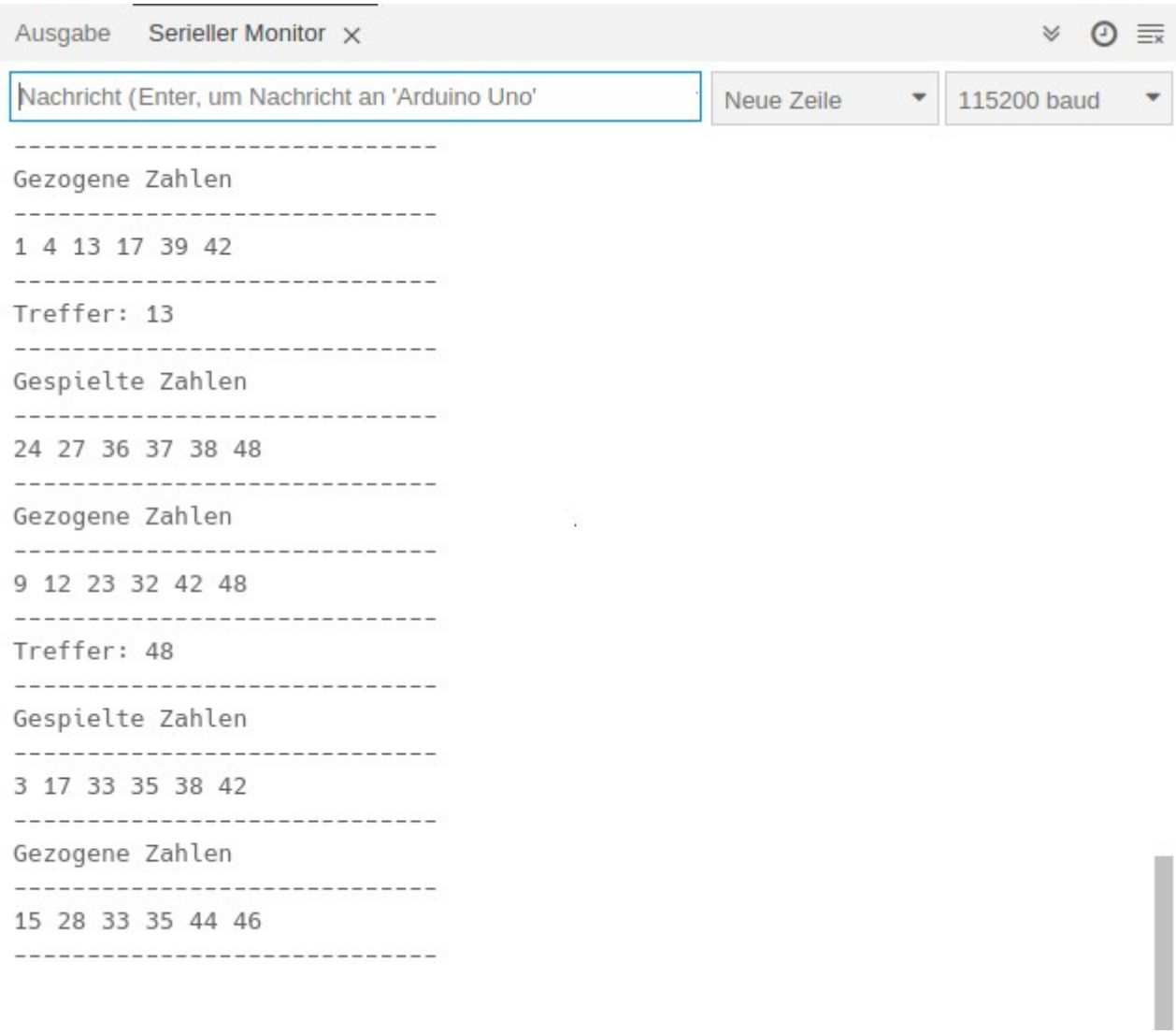
Der Arduino „spielt“ für dich 6 Zahlen. Anschließend werden die Lottozahlen gezogen, aufsteigend sortiert und auf die Anzahl der richtigen Zahlen untersucht.

So sieht es aus:



Für diese Aufgabe brauchst du nur den Arduino.

Definiere die Variablen:

```
// Array für die angekreuzten Zahlen
int SpielZahl[6];

// Array für die gespielten Zahlen und die Lottozahlen
int LottoZahl[6];

// Minimum/Maximum der Zufallszahlen
int Minimum = 1;
int Maximum = 49;
```

```
// Anzahl der zu ziehenden Zahlen
int Anzahl = 6;
```

Für die Bestimmung der Spielzahlen und die Ziehung der Lottozahlen wird jeweils eine Funktion verwendet:

```
int Ziehung()
{
  for (int i = 0; i < Anzahl; i ++)
  {
    /*
      die Lottozahl gilt solange als bereits gezogen
      bis in der for-Schleife nachgewiesen wird, dass sie neu ist
      und BereitsGezogen den Wert false hat
    */
    bool BereitsGezogen = true;

    while (BereitsGezogen)
    {
      // Zahl ziehen
      LottoZahl[i] = random(Minimum, Maximum);
      BereitsGezogen = false;
      /*
        Zufallszahl mit den bereits gezogenen Zahlen vergleichen
        i wird in der ersten for-Schleife hochgezählt
        alle bisherigen Zahlen (ii) werden mit den bisher gezogenen
        (außer der gerade gezogenen) verglichen (ii < i)
        ist sie identisch, erhält BereitsGezogen den Wert true
        es wird erneut eine Zufallszahl bestimmt
        wenn die Zahl neu ist, (BereitsGezogen hat noch den Wert false)
        wird die while-Schleife verlassen und die nächste Zahl gezogen
      */

      for (int ii = 0; ii < i; ii ++)
      {
        if (LottoZahl[i] == LottoZahl[ii]) BereitsGezogen = true;
      }
    }
  }

  /*
    das Array mit den Zahlen wird an das Hauptprogramm zurückgegeben
    beim ersten Aufruf sind es die gespielten Zahlen
    beim zweiten Aufruf die gezogenen Lottozahlen
  */
  return LottoZahl[Anzahl];
}
```

Das Sortieren der Zahlen erledigt die Methode ZahlenSortieren:

```
void ZahlenSortieren(int Zahlen[], int Groesse)
{
  // unsortierten Bereich des Arrays durchlaufen
  for (int i = 0; i < (Groesse - 1); i++)
  {
    // bei jedem Durchlauf wird das jeweils letzte Element weggelassen
    for (int ii = 0; ii < (Groesse - (i + 1)); ii++)
    {
      /*
        wenn die aktuelle Zahl größer als die nachfolgende Zahl ist
        -> aktuelle Zahl temporär speichern
        -> Zahlen vertauschen
        -> temporäre Zahl der nachfolgenden Zahl zuweisen
      */
      if (Zahlen[ii] > Zahlen[ii + 1])
      {
        int zwischengespeicherteZahl = Zahlen[ii];
        Zahlen[ii] = Zahlen[ii + 1];
        Zahlen[ii + 1] = zwischengespeicherteZahl;
      }
    }
  }
}
```

Der setup-Teil:

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

  // Zufallsgenerator starten
  randomSeed(analogRead(0));
  Serial.println("z = N\u00e4chste Ziehung");
}
```

Der loop-Teil beginnt mit dem Zurücksetzen der Strings und dem Zurücksetzen der Trefferanzahl. Mit SerialRead wird die Eingabe der Tastatur abgefragt. Dann werden die Lottozahlen gezogen.

```
void loop()
{
  String GespielteZahlen = "";
  String GezogeneZahlen = "";
  int Treffer = 0;

  SensorLesen = digitalRead(SENSOR);
  if (SensorLesen == HIGH)
  {
    delay(200);

    // Ziehung der "angekreuzten" Zahlen
    Ziehung();
```

Anschließend wird das Array der gespielten Zahlen mithilfe der Methode ZahlenSortieren aufsteigend sortiert und das Array als String zusammengesetzt und ausgegeben.

Innerhalb der for-Schleife wird das Array der gespielten Zahlen (SpielZahl) als String erzeugt. Es wird benötigt, um später die Anzahl der Treffer zu bestimmen.

```
    ZahlenSortieren(LottoZahl, 6);
    for (int i = 0; i < Anzahl; i ++)
    {
      GespielteZahlen = GespielteZahlen + LottoZahl[i] + " ";
      /*
         dem Array SpielZahl (die „gespielten“ Zahlen)
         wird das identische Element des Arrays LottoZahl zugewiesen
         Dieser Schritt darf für die Ziehung der Lottozahlen
         nicht wiederholt werden.
      */
      SpielZahl[i] = LottoZahl[i];
    }
```

Die Ziehung der Zahlen, das Sortieren der Zahlen und das Zusammensetzen des Strings GezogeneZahlen wird für die eigentlichen Lottozahlen wiederholt.
Jetzt sollen die die übereinstimmenden Zahlen ermittelt werden:

```
    // Vergleichen der Elemente der Arrays SpielZahl und LottoZahl
    for (int i = 0; i < Anzahl; i++)
    {
      for (int ii = 0; ii < Anzahl; ii++)
      {
        // Übereinstimmung gefunden -> Treffer um 1 erhöhen
        // AnzahlTreffer: String für übereinstimmende Zahlen erzeugen
        if (SpielZahl[i] == LottoZahl[ii])
        {
          AnzahlTreffer = AnzahlTreffer + SpielZahl[i] + " ";
          Treffer ++;
        }
      }
    }
```

Für die Anzeige wird aus den Elementen des Arrays LottoZahl ein String „zusammengebaut“.

```
    // String GezogeneZahlen aus den Elementen des Arrays LottoZahl
    // zusammensetzen
    for (int i = 0; i < Anzahl; i++)
    {
      GezogeneZahlen = GezogeneZahlen + LottoZahl[i] + " ";
    }
```

Die Anzeige im Seriellen Monitor:

```
    // Anzeige Serieller Monitor
    Serial.println("Gespielte Zahlen");
    Serial.println("----------------------------");
    Serial.println(GespielteZahlen);
    Serial.println("----------------------------");
    Serial.println("Gezogene Zahlen");
    Serial.println("----------------------------");
    Serial.println(GezogeneZahlen);
    Serial.println("----------------------------");

    // Treffer anzeigen
    if (Treffer == 0) Serial.println("keine Treffer!");
    else Serial.println("Treffer: " + AnzahlTreffer + "");
    Serial.println("----------------------------");
  }
}
```