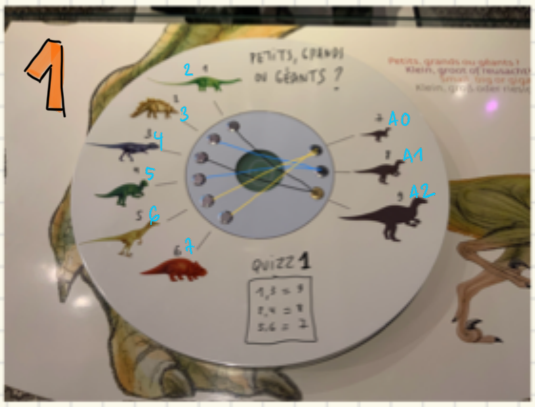
QUIZZ DINO

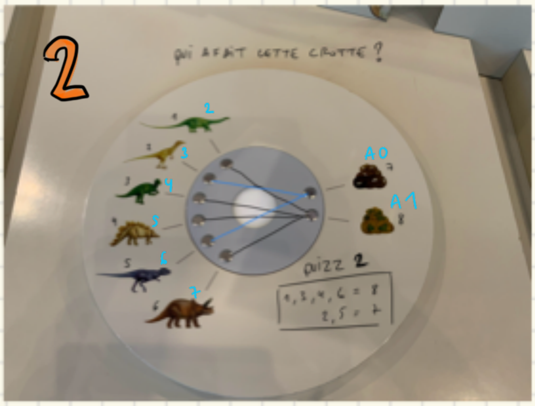


A0 = 6,7
A1 = 3,5
A2 = 2,4

PINS / SOLUTIONS

```
const int questions[] = {A0, A1, A2}; // Boutons des questions
const int reponses[] = {2, 3, 4, 5, 6, 7}; // Boutons des réponses

const int correspondances[][6] = {
  {-1, -1, -1, -1, 6, 7}, // Question 1
  {-1, 3, -1, 5, -1, -1} // Question 2
  {2, -1, 4, -1, -1, -1}, // Question 3
```

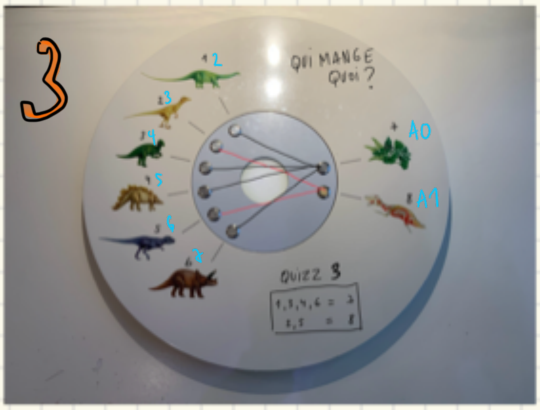


A0 = 3,6
A1 = 2,4,5,7

PINS / SOLUTIONS

```
const int questions[] = {A0, A1}; // Boutons des questions
const int reponses[] = {2, 3, 4, 5, 6, 7}; // Boutons des réponses

const int correspondances[][6] = {
  {-1, 3, -1, -1, 6, -1}, // Question 1
  {2, -1, 4, 5, -1, 7} // Question 2
```

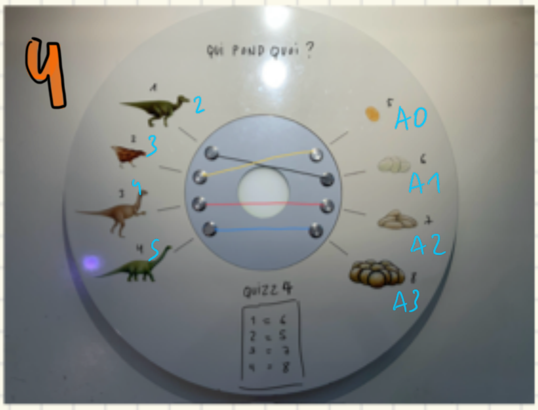


A0 = 2,4,5,7
A1 = 3,6

PINS / SOLUTIONS

```
const int questions[] = {A0, A1}; // Boutons des questions
const int reponses[] = {2, 3, 4, 5, 6, 7}; // Boutons des réponses

const int correspondances[][6] = {
  {2, -1, 4, 5, -1, 7} // Question 1
  {-1, 3, -1, -1, 6, -1}, // Question 2
```

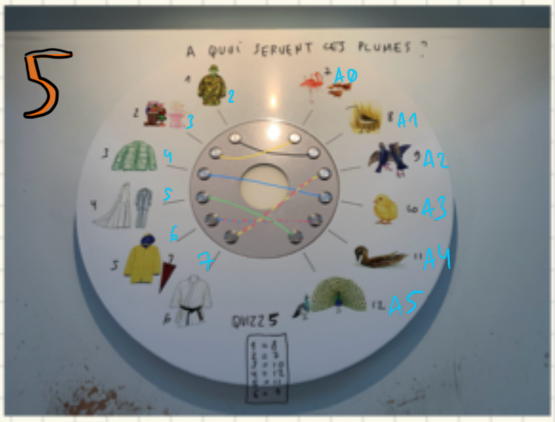


A0 = 3
A1 = 2
A2 = 4
A3 = 5

PINS / SOLUTIONS

```
const int questions[] = {A0, A1, A2, A3}; // Boutons des questions
const int reponses[] = {2, 3, 4, 5}; // Boutons des réponses

const int correspondances[][4] = {
  {-1, 3, -1, -1} // Question 1
  {2, -1, -1, -1}, // Question 2
  {-1, -1, 4, -1}, // Question 3
  {-1, -1, -1, 5}, // Question 4
```



A0 = 3	A1 = 2	A2 = 7	A3 = 4	A4 = 6	A5 = 5
QUESTION 1	QUESTION 2	QUESTION 3	QUESTION 4	QUESTION 5	QUESTION 6

```
const int questions[] = {A0, A1, A2, A3, A4, A5}; // Boutons des questions
const int reponses[] = {2, 3, 4, 5, 6, 7}; // Boutons des réponses

const int correspondances[][6] = {
  {-1, 3, -1, -1, -1, -1} // Question 1
  {2, -1, -1, -1, -1,-1}, // Question 2
  {-1, -1, -1, -1, -1, 7}, // Question 3
  {-1, -1, 4, -1, -1, -1}, // Question 4
  {-1, -1, -1, -1, 6, -1}, // Question 5
  {-1, -1, -1, 5, -1, -1}, // Question 6
```

CODE GÉNÉRAL

+ZONE À ADAPTER A CHAQUE QUIZZ

```
const int ledVerte = 10;
const int ledRouge = 11;

const int questions[] = {A2, 3}; // Boutons des questions
const int reponses[] = {5, 6, 7}; // Boutons des réponses

const int correspondances[][3] = {
  {5, 6, -1}, // Question 1 a deux bonnes réponses
  {7, -1, -1} // Question 2 a une seule bonne réponse
};

const int nbQuestions = sizeof(questions) / sizeof(questions[0]);
const int nbReponses = sizeof(reponses) / sizeof(reponses[0]);
const int maxReponses = sizeof(correspondances[0]) / sizeof(correspondances[0][0]);

void setup() {
  pinMode(ledVerte, OUTPUT);
  pinMode(ledRouge, OUTPUT);

  for (int i = 0; i < nbQuestions; i++) {
    pinMode(questions[i], INPUT_PULLUP);
  }
  for (int i = 0; i < nbReponses; i++) {
    pinMode(reponses[i], INPUT_PULLUP);
  }
}

void loop() {
  int questionAppuyee = -1;
  int reponseAppuyee = -1;
  int nbBoutonsAppuyes = 0;

  for (int i = 0; i < nbQuestions; i++) {
    if (digitalRead(questions[i]) == LOW) {
      questionAppuyee = i;
      nbBoutonsAppuyes++;
    }
  }

  for (int j = 0; j < nbReponses; j++) {
    if (digitalRead(reponses[j]) == LOW) {
      reponseAppuyee = reponses[j];
      nbBoutonsAppuyes++;
    }
  }

  if (nbBoutonsAppuyes == 2 && questionAppuyee != -1 && reponseAppuyee != -1) {
    bool bonneReponse = false;
    for (int k = 0; k < maxReponses; k++) {
      if (correspondances[questionAppuyee][k] == reponseAppuyee) {
        bonneReponse = true;
        break;
      }
    }
    if (bonneReponse) {
      digitalWrite(ledVerte, HIGH);
      digitalWrite(ledRouge, LOW);
    } else {
      digitalWrite(ledVerte, LOW);
      digitalWrite(ledRouge, HIGH);
    }
    delay(300);
    digitalWrite(ledVerte, LOW);
    digitalWrite(ledRouge, LOW);
  } else {
    digitalWrite(ledVerte, LOW);
    digitalWrite(ledRouge, LOW);
  }
}
```