```dart
import 'dart:math';

int binaryToDecimal(String binary) {
  int decimal = 0;
  int power = 0;

  for (int i = binary.length - 1; i >= 0; i--) {
    int digit = int.parse(binary[i]);

    decimal += digit * (1 << power);
    power++;
  }

  return decimal;
}

int octalToDecimal(String octal) {
  int decimal = 0;
  int power = 0;

  for (int i = octal.length - 1; i >= 0; i--) {
    int digit = int.parse(octal[i]);

    decimal += digit * pow(8, power).toInt();
    power++;
  }

  return decimal;
}
```

```dart
int hexadecimalToDecimal(String hexadecimal) {
  int decimal = 0;
  int power = 0;

  for (int i = hexadecimal.length - 1; i >= 0; i--) {
    String digit = hexadecimal[i];

    int digitValue;
    if (digit == 'A' || digit == 'a') {
      digitValue = 10;
    } else if (digit == 'B' || digit == 'b') {
      digitValue = 11;
    } else if (digit == 'C' || digit == 'c') {
      digitValue = 12;
    } else if (digit == 'D' || digit == 'd') {
      digitValue = 13;
    } else if (digit == 'E' || digit == 'e') {
      digitValue = 14;
    } else if (digit == 'F' || digit == 'f') {
      digitValue = 15;
    } else {
      digitValue = int.parse(digit);
    }

    decimal += digitValue * pow(16, power).toInt();
    power++;
  }

  return decimal;
}

String decimalToBinary(int decimal) {
  if (decimal == 0) {
    return '0';
  }

  String binary = '';

  while (decimal > 0) {
    int remainder = decimal % 2;
    binary = binary + remainder.toString();
    decimal = decimal ~/ 2;
  }

  return binary.split('').reversed.join('');
}

String binaryToOctal(String binary) {
  int decimal = binaryToDecimal(binary);
  String octal = decimalToOctal(decimal);
  return octal;
}
```

```dart
String binaryToHexadecimal(String binary) {
  int decimal = binaryToDecimal(binary);
  String hexadecimal = decimalToHexadecimal(decimal);
  return hexadecimal;
}

String decimalToOctal(int decimal) {
  String octal = '';

  if (decimal == 0) {
    octal = '0';
  } else {
    while (decimal > 0) {
      int remainder = decimal % 8;
      octal = remainder.toString() + octal;
      decimal = decimal ~/ 8;
    }
  }

  return octal;
}

String decimalToHexadecimal(int decimal) {
  String hexadecimal = '';

  if (decimal == 0) {
    hexadecimal = '0';
  } else {
    while (decimal > 0) {
      int remainder = decimal % 16;
      String hexDigit;

      if (remainder < 10) {
        hexDigit = remainder.toString();
      } else {
        hexDigit = String.fromCharCode(65 + remainder - 10);
      }

      hexadecimal = hexDigit + hexadecimal;
      decimal = decimal ~/ 16;
    }
  }

  return hexadecimal;
}

String octalToBinary(String octal) {
  int decimal = octalToDecimal(octal);
  String binary = decimalToBinary(decimal);
  return binary;
}
```

```
String hexadecimalToBinary(String hexadecimal) {
  int decimal = hexadecimalToDecimal(hexadecimal);
  String binary = decimalToBinary(decimal);
  return binary;
}
```