

```

def convert_base(number, from_base, to_base):
    if from_base == 10:
        num_in_base_10 = int(number)
    else:
        num_in_base_10 = int(str(number), from_base)

    if to_base == 10:
        result = num_in_base_10
    else:
        result = ""
        while num_in_base_10 > 0:
            digit = num_in_base_10 % to_base
            if digit < 10:
                result = str(digit) + result
            else:
                result = chr(ord('A') + digit - 10) + result
            num_in_base_10 //= to_base
    return result

def main():
    try:
        number = input("Ingrese el número a convertir: ")
        from_base = int(input("Ingrese la base de origen (2-16): "))
        to_base = int(input("Ingrese la base de destino (2-16): "))

        if from_base < 2 or from_base > 16 or to_base < 2 or to_base > 16:
            print("Las bases deben estar entre 2 y 16.")
            return

        converted_number = convert_base(number, from_base, to_base)

        print("Número convertido:", converted_number)

        binary_number = convert_base(number, from_base, 2)
        ascii_message = ''.join([chr(int(binary_number[i:i+8], 2)) for i in range(0,
len(binary_number), 8)])

        print("Mensaje ASCII:", ascii_message)

    except ValueError:
        print("Entrada inválida. Asegúrate de ingresar números y bases válidos.")

if __name__ == "__main__":
    main()

```

