Clasificación y Extracción de Cadenas Biológicas en C++

Autor: Jayan Michael Caceres Cuba

16 de mayo de 2025

1. Introducción

Este informe describe el funcionamiento de dos funciones escritas en C++ diseñadas para trabajar con cadenas que representan secuencias biológicas: ADN, ARN o proteínas. El objetivo es clasificar dichas cadenas y, en caso de tratarse de una proteína, extraer los nombres completos de los aminoácidos.

2. Clasificación de cadenas

La función `clasificarCadena` permite identificar si una cadena representa una secuencia de ADN, ARN o una proteína, comparando los caracteres presentes con conjuntos predefinidos.

Código fuente

```
TipoCadena clasificarCadena(const std::string& cadena) {
    std::unordered_set<char> adn = { 'A', 'C', 'G', 'T' };
    std::unordered_set<char> arn = { 'A', 'C', 'G', 'U' };
    std::unordered_set<char> aminoacidos = { 'A', 'C', 'D', 'E', 'F', 'G', 'H',
                                             'I', 'K', 'L', 'M', 'N', 'P', 'Q',
                                             'R', 'S', 'T', 'V', 'W', 'Y' };

    std::unordered_set<char> caracteres;
    for (char c : cadena) {
        if (std::isalpha(c))
            caracteres.insert(std::toupper(c));
    }

    if (std::all_of(caracteres.begin(), caracteres.end(), [&](char c) { return adn.
        count(c); }))
        return TipoCadena::ADN;

    if (std::all_of(caracteres.begin(), caracteres.end(), [&](char c) { return arn.
        count(c); }))
        return TipoCadena::ARN;

    if (std::all_of(caracteres.begin(), caracteres.end(), [&](char c) { return
        aminoacidos.count(c); }))
        return TipoCadena::PROTEINA;

    return TipoCadena::DESCONOCIDA;
}
```

Listing 1: Función clasificarCadena

3. Extracción de aminoácidos

La función `extraerAminoacidos` convierte una cadena con letras correspondientes a aminoácidos en sus respectivos nombres completos.

Código fuente

```
std::vector<std::string> extraerAminoacidos(const std::string& cadena) {
    std::unordered_map<char, std::string> mapa = {
        {'A', "Alanina"},   {'C', "Ciste na"}, {'D', "Aspartato"}, {'E', "Glutamato"},
        {'F', "Fenilalanina"}, {'G', "Glicina"}, {'H', "Histidina"}, {'I', "Isoleucina"
            },
        {'K', "Lisina"},    {'L', "Leucina"},  {'M', "Metionina"},  {'N', "Asparagina"
            },
        {'P', "Prolina"},   {'Q', "Glutamina"}, {'R', "Arginina"},  {'S', "Serina"},
        {'T', "Treonina"},  {'V', "Valina"},    {'W', "Tript fano"}, {'Y', "Tirosina"}
    };

    std::vector<std::string> resultado;
    for (char c : cadena) {
        c = std::toupper(c);
        if (mapa.count(c))
            resultado.push_back(mapa[c]);
    }
    return resultado;
}
```

Listing 2: Función extraerAminoacidos

4. Pruebas unitarias

Para verificar el correcto funcionamiento de las funciones, se utilizaron pruebas unitarias mediante Google Test. A continuación se muestran las pruebas que validan la función `clasificarCadena`:

```
TEST(ClasificacionTest, EsADN) {
    EXPECT_EQ(clasificarCadena("ACGT"), TipoCadena::ADN);
}

TEST(ClasificacionTest, EsARN) {
    EXPECT_EQ(clasificarCadena("ACGU"), TipoCadena::ARN);
}

TEST(ClasificacionTest, EsProteina) {
    EXPECT_EQ(clasificarCadena("ACDEFGHIKLMNPQRSTVWY"), TipoCadena::PROTEINA);
}

TEST(ClasificacionTest, EsDesconocida) {
    EXPECT_EQ(clasificarCadena("XYZ123"), TipoCadena::DESCONOCIDA);
}
```

Listing 3: Pruebas unitarias con Google Test

Explicación

- Se comprueba que cadenas como .ªCGT"sean reconocidas como ADN.

- .ACGU.es identificada correctamente como ARN.
- Una secuencia de todos los aminoácidos estándar se clasifica como proteína.
- Una cadena con caracteres inválidos o no biológicos se clasifica como desconocida.

5. Conclusión

Las funciones implementadas permiten identificar de forma confiable el tipo de cadena biológica que se analiza, y en el caso de proteínas, extraer sus componentes en forma legible. Las pruebas unitarias confirman su funcionamiento y robustez.