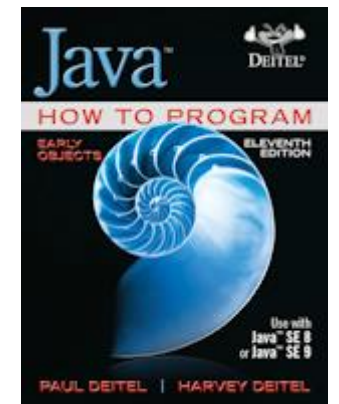


Guia de Laboratório

Lab 02

Prof. André Bernardi

andrebernardi@unifei.edu.br





2º Laboratório ECOP15 – 13 e 14 de setembro 2021

- **Utilizar como referência os exemplos:** Cap 14 páginas 473 a 488, 10ª ed Java como Programar.
 - StringMiscellaneous.java;
 - StringCompare.java;
 - SubString.java;
 - StringMiscellaneous2.java;



1ª Tarefa:

Crie um programa em Java, para testar a classe e identificar cada aminoácido de uma string de RNA, que devem estar implementados como descritos a seguir, nos itens a e b:



Item a:

- Escreva uma classe **StringRNA** que seja capaz de processar uma String de RNA. Strings de RNA são Strings que são formadas exclusivamente pelos caracteres 'A', 'C', 'G' e 'U' - nenhum outro caractere é permitido. Esta classe deve encapsular uma String e conter ao menos os seguintes métodos:



Item a:

- **Construtor**, que recebe uma String normal como argumento e copia-a para a String encapsulada pela classe se a passada por argumento estiver dentro dos parâmetros exigidos (isto é, se a String passada for "CAUGAUUAG", a String encapsulada será "CAUGAUUAG", mas se a String passada for "JAVA", a String encapsulada será vazia);
- **toString()**, que retorna a String encapsulada;
- **charAt()**, que retorna o caractere na posição que for passada como argumento;
- **length()**, que retorna o comprimento da String encapsulada.
- **quantosA()**, **quantosC()**, **quantosG()** e **quantosU()**, que retornam, respectivamente, quantos caracteres 'A', 'C', 'G' e 'U' existem na String encapsulada,



Exemplo de solução – letra a)

```
// Exemplo de solução da atividade
public class StringRNA{
{
    private String rna;

    // Construtor da classe
    public StringRNA( String s)
    {
        s = s.toUpperCase();
        if ( validaString(s) )
            rna = s;
        else
            rna = "";
    }
}
```

```

// retorna o char referente ao índice
public char charAt(int indice)
{
    return rna.charAt(indice);
}

// retorna somente a string interna
public String toString()
{
    return rna;
}

// retorna o numero de caracteres na cadeia
public long length ()
{
    return rna.length();
}

// retorna o numero de caracteres A na cadeia
public long quantosA()
{
    return quantosChar('A');
}

```

```

// retorna o numero de caracteres C na cadeia
public long quantosC()
{
    return quantosChar('C');
}

// retorna o numero de caracteres G na cadeia
public long quantosG()
{
    return quantosChar('G');
}

// retorna o numero de caracteres U na cadeia
public long quantosU()
{
    return quantosChar('U');
}

// retorna o numero de caracteres U na cadeia
private long quantosChar(char teste){
    long cont = 0;
    for (long i=0; i< rna.length(); i++)
        if ( rna.charAt(i) == teste)
            cont++;
    return cont;
}

```

```
// verifica se a String contem somente os caracteres permitidos
private boolean validaString(String validar)
{
    for(int i=0; i< validar.length(); i++) {
        if( !((validar.charAt(i) == 'A') || (validar.charAt(i) == 'C') ||
            (validar.charAt(i) == 'G') || (validar.charAt(i) == 'U')) )
            return false;
    }
    return true;
}

public boolean setString( String s)
{
    s = s.toUpperCase();
    if ( validaString(s) ){
        rna = s;
        return true;
    }
    else {
        rna = "";
        return false;
    }
}
```



```

// main de teste da classe
public static void main(String [] a)
{
    java.util.Scanner input = new java.util.Scanner (System.in);
    System.out.println(" Digite uma cadeia de RNA: ");
    String st = input.next();

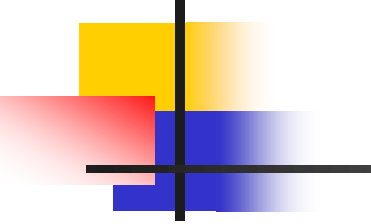
    StringRNA str = new StringRNA(st);
    //str.setString(st);
    if(str.equals(""))
        System.out.println("Cadeia de caracteres inválida! ");
    else{
        System.out.println("A cadeia de caracteres digitada foi: " + str);
        System.out.println("Ela possui : " + str.quantosA() + " letras A.");
        System.out.println("Ela possui : " + str.quantosC() + " letras C.");
        System.out.println("Ela possui : " + str.quantosG() + " letras G.");
        System.out.println("Ela possui : " + str.quantosU() + " letras U.");
        System.out.println("O primeiro caractere na cadeia de RNA e:" + str.charAt(0));
    }
}
} // fim da classe

```



Item b:

Aminoácidos são definidos por conjuntos de três caracteres em Strings de RNA, sendo que cada aminoácido pode ter mais do que um conjunto de três caracteres correspondentes. Existem vinte aminoácidos, mas algumas combinações de três caracteres formam um *senal de término*. Os vinte aminoácidos e o sinal de término, seus símbolos e as combinações correspondentes são:



Aminoácidos	Símbolos	Combinações
Alanine	A	GCU, GCC, GCA e GCG
Arginine	R	CGU, CGC, CGA, CGG, AGA e AGG
Aspartic Acid	D	GAU e GAC
Arparginine	N	AAU e AAC
Cysteine	C	UGU e UGC
Glutamic acid	E	GAA e GAG
Glutamine	Q	CAA e CAG
Glycine	G	GGU, GGC, GGA e GGG
Histine	H	CAU e CAC
Isoleucine	I	AUU, AUC e AUA
Leucine	L	UUA, UUG, CUU, CUC, CUA e CUG
Lysine	K	AAA e AAG
Methionine	M	AUG
Phenylalanine	F	UUU e UUC
Proline	P	CCU, CCC, CCA e CCG
Serine	S	AGU, AGC, UCU, UCC, UCA e UCG
Threonine	T	ACU, ACC, ACA e ACG
Tryptophan	W	UGG
Tyrosine	X	UAU e UAC
Valine	V	GUU, GUC, GUA e GUG
Sinais de término	.	UAA, UAG e UGA



Item b:

Considerando a tabela dada:

- Escreva na classe StringRNA um método que a partir da String contendo RNA encapsulada na classe, percorra a String de três em três caracteres e retorne uma String com os nomes dos aminoácidos correspondentes.
- Escreva também um método que percorra a String de três em três caracteres e ao final retorne uma outra String contendo os símbolos dos aminoácidos: se a String encapsulada fosse "CUGUAAGGGCCA" este método retornaria "L.GP".

Ambos os métodos devem desconsiderar restos de Strings que não formem três caracteres: uma String de oito caracteres corresponderá à dois aminoácidos e dois caracteres sobrarão.



Exemplo de solução – letra b)

```
// Exemplo de solução da atividade b
// - isolar três caracteres da String interna em uma substring
// - percorrer a tabela para encontrar correspondência.
// Utilizar o acesso aos elementos da tabela, para resolver as duas funções propostas.
//for (int i=0; i< tabela.lenght; i++){ //percorre as linhas
//  for(int j= 2; j< tabela[i].lenght; j++) // a partir do índice 2 que contem as bases.
//  {
//      if(tabela[i][j].equals(subs))  {
//          // encontrou correspondência
//          tabela[i][0]  ; // nome do aminoacido
//          tabela[i][1]  ; // sigla do aminoacido
//      }
//  }
//}
```

```
// tabela com os nomes dos aminoácidos e códigos
```

```
private String tabela[][] = {  
    {"Alanine",          "A",  "GCU", "GCC", "GCA" , "GCG"           },  
    {"Arginine",         "R",  "CGU", "CGC", "CGA", "CGG", "AGA", "AGG"       },  
    {"Aspartic Acid",    "D",  "GAU", "GAC"                               },  
    {"Arparagine",       "N",  "AAU", "AAC"                               },  
    {"Cysteine",         "C",  "UGU", "UGC"                               },  
    {"Glutamic acid",    "E",  "GAA", "GAG"                               },  
    {"Glutamine",        "Q",  "CAA", "CAG"                               },  
    {"Glycine",          "G",  "GGU", "GGC", "GGA" , "GGG"           },  
    {"Histine",          "H",  "CAU", "CAC"                               },  
    {"Isoleucine",       "I",  "AUU", "AUC", "AUA"                               },  
    {"Leucine",          "L",  "UUA", "UUG", "CUU", "CUC", "CUA" , "CUG"       },  
    {"Lysine",           "K",  "AAA", "AAG"                               },  
    {"Methionine",       "M",  "AUG"                               },  
    {"Phenylalanine",    "F",  "UUU", "UUC"                               },  
    {"Proline",          "P",  "CCU", "CCC", "CCA" , "CCG"           },  
    {"Serine",           "S",  "AGU", "AGC", "UCU" , "UCC", "UCA" , "UCG"       },  
    {"Threonine",        "T",  "ACU", "ACC", "ACA" , "ACG"           },  
    {"Tryptophan",       "W",  "UGG"                               },  
    {"Tyrosine",         "X",  "UAU", "UAC"                               },  
    {"Valine",           "V",  "GUU", "GUC", "GUA" , "GUG"           },  
    {"Sinais de término", ". ", "UAA", "UAG", "UGA"           } };
```