

Universidade Federal de Ouro Preto Departamento de Computação - DECOM BCC241 – Projeto e Análise de Algoritmos Prof. Anderson Almeida Ferreira Exercícios – Subsequência Palíndromo

Turma: 11

Nome: Pedro Lucas Damasceno Silva

Matrícula: 20.1.4003

```
1. Algoritmo Recursivo
```

```
MSP(s[n], begin, end) {
 if (begin > end) {
   return 0;
 if (begin == end) {
   return 1;
 if (s[begin] == s[end]) {
   return 2 + MSP_Rec(s, begin+1, end-1);
 }
 else {
   return max(MSP_Rec(s, begin, end-1), MSP_Rec(s, begin+1, end));
 }
}
2. Algoritmo de Memorização
MSP(s[n], m[n][n], begin, end) {
 if (begin > end) {
   return 0;
 }
 if (begin == end) {
    return 1;
 if (!isset(m[begin, end])) {
   if (s[begin] == S[end]) {
      m[begin, end] = 2 + MSP(s, m, begin+1, end-1);
   }
      m[begin, end] = max(MSP(s, m, begin, end-1), MSP(s, m, begin+1, end));
  }
  return m;
```



Universidade Federal de Ouro Preto Departamento de Computação - DECOM BCC241 – Projeto e Análise de Algoritmos Prof. Anderson Almeida Ferreira Exercícios – Subsequência Palíndromo

3. Algoritmo Iterativo

```
MSP(s[n], m[n][n]) \{
  for (i = inicio, i < fim; i++) {
    m[i, i] = -1;
  for (tam = 2; i < n; tam++) {
    for (i = 1; i < n - tam + 1; i++) {
      j = i + tam - 1;
      if (s[i] == s[j] \&\& tam == 2) {
        m[i,j] = 2;
      }
      else if (s[i] == s[j]) {
        m[i, j] = 2 + m[i+1, j-1];
      }
      else {
        m[i, j] = max(m[i, j-1], m[i+1, j]);
    }
  }
  return m;
}
```