Design & Analysis Of Algorithms

Lab 5

S Abhishek

AM.EN.U4CSE19147

Colab Link

Karatsuba Algorithm

```
def Karatsuba(x, y):

  if len(str(x)) == 1 or len(str(y)) == 1:

    return x * y

  l = max(len(str(x)), len(str(y)))

  length = l // 2

  x1 = x // 10 ** length
  x2 = x % 10 ** length

  y1 = y // 10 ** length
  y2 = y % 10 ** length

  x1_y1 = Karatsuba(x1, y1)
  x2_y2 = Karatsuba(x2, y2)

  sum = Karatsuba(x1 + x2, y1 + y2) - x1_y1 - x2_y2

  return x1_y1 * (10 ** ( 2 * length)) + sum * (10 ** length) + x2_y2


print("\nAnswer :", Karatsuba(int(input("Enter the Number 1 : ")),
int(input("\nEnter the Number 2 : "))))
```

```
Enter the Number 1 : 1050

Enter the Number 2 : 2050

Answer : 2152500
```

Strassen Algorithm

```
Matrix A = | a b |
           | c d |

Matrix B = | e f |
           | g h |

P1 = a * (f - h)

P2 = (a + b) * h

P3 = (c + d) * e

P4 = d * (g - e)

P5 = (a + d) * (e + h)

P6 = (b - d) * (g + h)

P7 = (a - c) * (e + f)

matrix C = | P5 + P4 - P2 + P6         P1 + P2     |
           |   P3 + P4           P1 + P5 - P3 - P7 |
```

```python
def add(m1, m2):

  if type(m1) == int:

    return m1 + m2

  else:

    n = len(m1)

    return [[m1[j][i] + m2[j][i] for i in range(n)] for j in range(n)]
```

```
def sub(m1, m2):
  if type(m1) == int:
    return m1 - m2
  else:
    n = len(m1)
    return [[m1[j][i] - m2[j][i] for i in range(n)] for j in range(n)]


def Strassen(m1, m2):
  n = len(m1)
  half = n // 2
  if n == 1:
    return m1[0][0] * m2[0][0]
  A = [[m1[i][j] for j in range(half)] for i in range(half)]
  B = [[m1[i][j] for j in range(half, n)] for i in range(half)]
  C = [[m1[i][j] for j in range(half)] for i in range(half, n)]
  D = [[m1[i][j] for j in range(half, n)] for i in range(half, n)]

  E = [[m2[i][j] for j in range(half)] for i in range(half)]
  F = [[m2[i][j] for j in range(half, n)] for i in range(half)]
  G = [[m2[i][j] for j in range(half)] for i in range(half, n)]
  H = [[m2[i][j] for j in range(half, n)] for i in range(half, n)]

  P1 = Strassen(A, sub(F, H))
  P2 = Strassen(add(A, B), H)
  P3 = Strassen(add(C, D), E)
  P4 = Strassen(D, sub(G, E))
  P5 = Strassen(add(A, D), add(E, H))
  P6 = Strassen(sub(B, D), add(G, H))
  P7 = Strassen(sub(A, C), add(E, F))

  LU = add(sub(add(P5, P4), P2), P6)
  LL = add(P3, P4)
  RU = add(P1, P2)
  RL = sub(sub(add(P1, P5), P3), P7)
```

```
    if n > 2:

        m = [[ 0 for i in range(n)] for j in range(n)]

        for i in range(half):

          for j in range(half):

            m[i][j] = LU[i][j]
            m[i + half][j] = LL[i][j]

            m[i][j + half] = RU[i][j]
            m[i + half][j + half] = RL[i][j]

        return m

      else:

        return [[LU, RU], [LL, RL]]




m1 = [(1,2,3,4),
      (5,6,7,8),
      (9,10,11,12),
      (13,14,15,16)]

m2 = [(21,22,23,24),
      (25,26,27,28),
      (29,30,31,32),
      (33,34,35,36)]

m = Strassen(m1, m2)


for i in range(len(m1)):

  for j in range(len(m1[0])):

    print(m[i][j], end = "   ")

  print()
```

```
290   300   310   320
722   748   774   800
1154   1196   1238   1280
1586   1644   1702   1760
```

Thankyou!!