

Praktikum Algoritma Pemrograman

Modul 9 - List2D

Sabtu, 27 Desember 2020

Tujuan dari modul ini agar mahasiswa lebih memahami konsep dan implementasi dari **List 2D**. Kerjakan tugas-tugas yang terdapat dalam modul ini, dengan ketentuan sebagai berikut :

1. Semua jawaban modul dikerjakan dalam format *.ipynb
2. Penamaan file ipynb adalah : NPM_ModulX_TopikModul.ipynb, misalkan, 200411100077_Modul9_List2D.ipynb
3. Print menjadi file pdf, dokumen ipynb tersebut dengan nama yang sama, hanya saja berekstensi pdf, misalkan, 200411100077_Modul9_List2D.pdf
4. Submit **link collaboratory** yang berisi file ipynb tersebut, dan submit **file pdf**
5. Kejujuran selalu jadi yang utama, kerjakan sendiri, tidak diperkenankan plagiarism

1 Fungsi Rekursif

Tulis ringkasan atau penjelasan hal-hal berikut, dengan kata-kata kalian sendiri :

1. Jelaskan mengenai List 2D, Perbedaannya dengan List 1D
2. Berikan contoh data yang menggunakan representasi List2D

2 Implementasi-Modul Matriks

Buatlah modul matriks yang berisi proses pembuatan dan pengolahan data list 2D, sebagai berikut

2.1 *Create Matrix 2D*

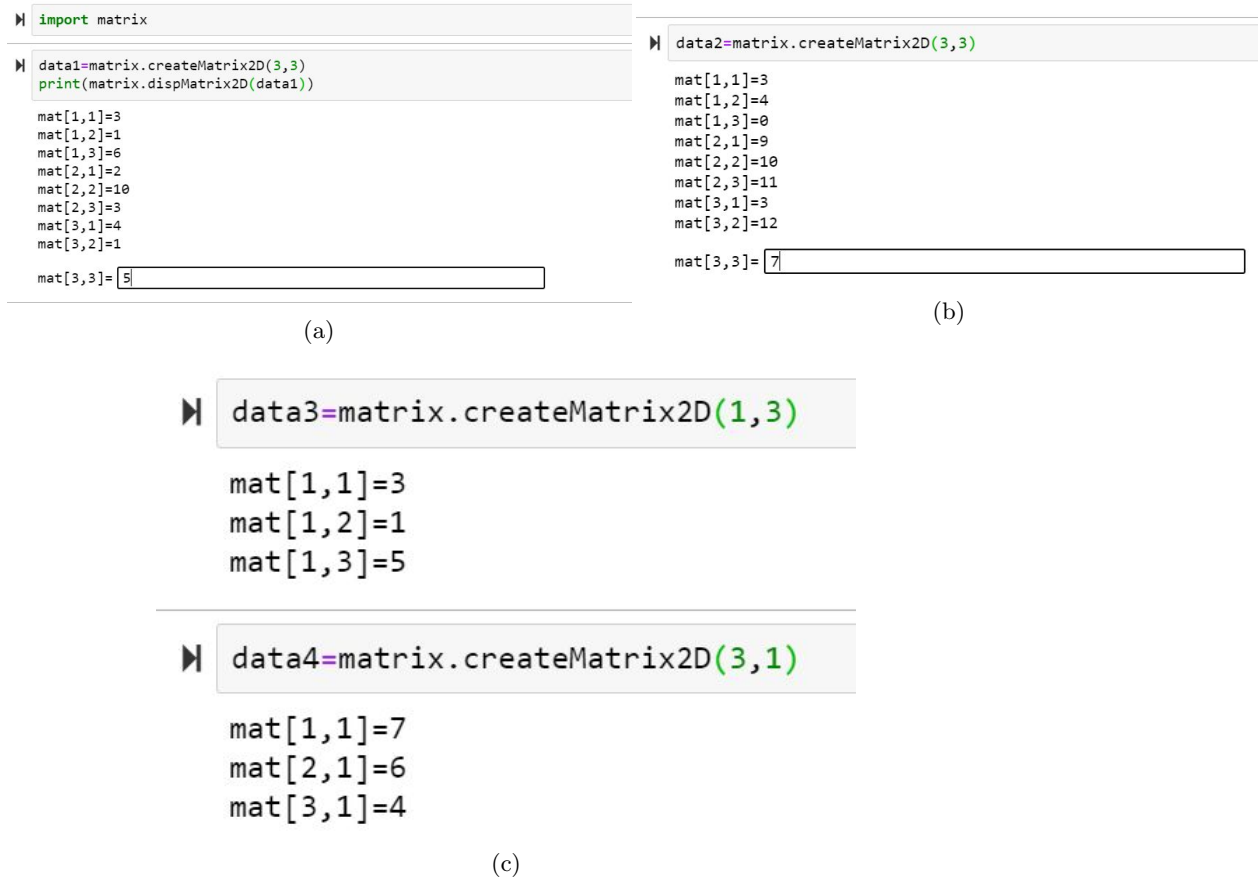
Buatlah **fungsi create Matrix2D** untuk membuat suatu matriks 2D dengan representasi list 2D (dengan argument atau parameter adalah jumlah baris dan jumlah kolom), seperti yang terlihat pada Gambar 1.

2.2 *Display Matrix*

Buatlah **fungsi display Matrix2D** untuk menampilkan suatu matriks 2D (dengan argument atau parameter adalah matriks yang akan ditampilkan), seperti yang terlihat pada Gambar 2.

2.3 *Square Matrix*

Buatlah **fungsi is square Matrix2D** untuk mengecek apakah suatu matrix 2D adalah matriks square ? (dengan argument atau parameter adalah matriks yang akan dicek), seperti yang terlihat pada Gambar 3.



Gambar 1: Create Matrix2D, argument : Jumlah Baris dan Jumlah Kolom

2.4 Penjumlahan Matriks

Buatlah **fungsi Penjumlahan Matrix2D** untuk menjumlahkan dua buah matriks (dengan argument atau parameter adalah matriks yang akan dijumlahkan). Di dalam fungsi tersebut dilakukan pengecekan apakah dua buah matriks dapat dijumlahkan, seperti yang terlihat pada Gambar 4.

2.5 Perkalian Matriks

Buatlah **fungsi Perkalian Matrix2D** untuk mengalikan dua buah matriks (dengan argument atau parameter adalah matriks yang akan dijumlahkan). Di dalam fungsi tersebut dilakukan pengecekan apakah dua buah matriks dapat dikalikan, seperti yang terlihat pada Gambar 5.

2.6 Nilai Maksimal Matriks

Buatlah **fungsi Maksimal Matrix2D** untuk mencari nilai maksimal, dengan ketentuan :

1. parameter kedua adalah 'baris', maka akan dihasilkan nilai maksimal dari setiap baris dari suatu matriks
2. parameter kedua adalah 'kolom', maka akan dihasilkan nilai maksimal dari setiap kolom dari suatu matriks
3. parameter kedua adalah '*', maka akan dihasilkan nilai maksimal dari suatu matriks

```

▶ data1=matrix.createMatrix2D(3,3)
print(matrix.dispMatrix2D(data1))

```

```

mat[1,1]=3
mat[1,2]=1
mat[1,3]=6
mat[2,1]=2
mat[2,2]=10
mat[2,3]=3
mat[3,1]=4
mat[3,2]=1
mat[3,3]=5
| 3 1 6 |
| 2 10 3 |
| 4 1 5 |

```

(a)

```

▶ print('data 1 :',matrix.dispMatrix2D(data1),sep='\n')
print('data 2 :',matrix.dispMatrix2D(data2),sep='\n')
print('data 3 :',matrix.dispMatrix2D(data3),sep='\n')
print('data 4 :',matrix.dispMatrix2D(data4),sep='\n')

```

```

data 1 :
| 3 1 6 |
| 2 10 3 |
| 4 1 5 |

data 2 :
| 3 4 0 |
| 9 10 11 |
| 3 12 7 |

data 3 :
| 3 1 5 |

data 4 :
| 7 |
| 6 |
| 4 |

```

(b)

Gambar 2: Display Matrix

Contoh output dari fungsi ini dapat dilihat pada Gambar 6

Selamat Mengerjakan, Selalu Latihan, Jujur
 harus dimulai kapanpun, Bertanya jika kurang
 mengerti, #StayAtHome,
 #LearningFromHome

Algoritma Pemrograman
Indah Agustien Siradjuddin

```

▶ print('data 1 :',matrix.dispMatrix2D(data1),sep='\n')
print('issquare ? ', matrix.isSquare(data1),'\n')
print('data 2 :',matrix.dispMatrix2D(data2),sep='\n')
print('issquare ? ', matrix.isSquare(data2),'\n')
print('data 3 :',matrix.dispMatrix2D(data3),sep='\n')
print('issquare ? ', matrix.isSquare(data3),'\n')
print('data 4 :',matrix.dispMatrix2D(data4),sep='\n')
print('issquare ? ', matrix.isSquare(data4),'\n')

```

```

data 1 :
| 3 1 6 |
| 2 10 3 |
| 4 1 5 |

```

```

issquare ? True

```

```

data 2 :
| 3 4 0 |
| 9 10 11 |
| 3 12 7 |

```

```

issquare ? True

```

```

data 3 :
| 3 1 5 |

```

```

issquare ? False

```

```

data 4 :
| 7 |
| 6 |
| 4 |

```

```

issquare ? False

```

Gambar 3: isSquareMatrix ?

```

▶ jumlah=matrix.addMatrix(data1,data2)
print('data 1 :',matrix.dispMatrix2D(data1),sep='\n')
print('data 2 :',matrix.dispMatrix2D(data2),sep='\n')
print('Jumlah :',matrix.dispMatrix2D(jumlah),sep='\n')

```

```

data 1 :
| 3 1 6 |
| 2 10 3 |
| 4 1 5 |

```

```

data 2 :
| 3 4 0 |
| 9 10 11 |
| 3 12 7 |

```

```

Jumlah :
| 6 5 6 |
| 11 20 14 |
| 7 13 12 |

```

(a)

```

▶ matrix.addMatrix(data1,data3)

```

```

]: 'Ukuran Tidak Sama'

```

(b)

Gambar 4: Penjumlahan Matriks

```

▶ hasilKali=matrix.multMat(data1,data2)
print('data 1 :',matrix.dispMatrix2D(data1),sep='\n')
print('data 2 :',matrix.dispMatrix2D(data2),sep='\n')
print('Hasil Perkalian :',matrix.dispMatrix2D(hasilKali),sep='\n')

```

```

data 1 :
| 3 1 6 |
| 2 10 3 |
| 4 1 5 |

```

```

data 2 :
| 3 4 0 |
| 9 10 11 |
| 3 12 7 |

```

```

Hasil Perkalian :
| 36 94 53 |
| 105 144 131 |
| 36 86 46 |

```

(a)

```

▶ hasilKali=matrix.multMat(data1,data3)
print(hasilKali)

```

Ukuran tidak Memenuhi

(b)

```

▶ hasilKali=matrix.multMat(data3,data4)
print('data 3 :',matrix.dispMatrix2D(data3),sep='\n')
print('data 4 :',matrix.dispMatrix2D(data4),sep='\n')
print('Hasil Perkalian :',matrix.dispMatrix2D(hasilKali),sep='\n')

```

```

data 3 :
| 3 1 5 |

```

```

data 4 :
| 7 |
| 6 |
| 4 |

```

```

Hasil Perkalian :
| 47 |

```

(c)

Gambar 5: Perkalian Matriks

```

▶ maksBaris=matrix.maksMat(data1,'baris')
maksKolom=matrix.maksMat(data1,'kolom')
maks=matrix.maksMat(data1,'*')
print('data 1 :',matrix.dispMatrix2D(data1),sep='\n')
print('Maksimum - Baris:',matrix.dispMatrix2D(maksBaris),sep='\n')
print('Maksimum - Kolom:',matrix.dispMatrix2D(maksKolom),sep='\n')
print('Maksimum :',maks)

```

```

data 1 :
| 3   1   6 |
| 2  10   3 |
| 4   1   5 |

```

```

Maksimum - Baris:
| 6 |
| 10 |
| 5 |

```

```

Maksimum - Kolom:
| 4   10   6 |

```

```

Maksimum : 10

```

(a)

```

▶ maksBaris=matrix.maksMat(data2,'baris')
maksKolom=matrix.maksMat(data2,'kolom')
maks=matrix.maksMat(data2,'*')
print('data 2 :',matrix.dispMatrix2D(data2),sep='\n')
print('Maksimum - Baris:',matrix.dispMatrix2D(maksBaris),sep='\n')
print('Maksimum - Kolom:',matrix.dispMatrix2D(maksKolom),sep='\n')
print('Maksimum :',maks)

```

```

data 2 :
| 3   4   0 |
| 9  10  11 |
| 3  12   7 |

```

```

Maksimum - Baris:
| 4 |
| 11 |
| 12 |

```

```

Maksimum - Kolom:
| 9   12   11 |

```

```

Maksimum : 12

```

(b)

Gambar 6: Nilai Maksimal Matriks