Tugas Pemograman Fundamental

***5 Algoritma, Flowchart, Beserta Source Code***

M. Farhan Aulia Pratama

12250113521

Kelas TIF-C’22

*source code tugas bisa dilihat di :*

*https://github.com/MFarhanZ1/KuliahJava*

***01.* Algoritma Mencari Kelajuan atau Kecepatan Seseorang Saat Berkendara!**

# Kasus

Ada seorang pembalap motor liar, ia berhasil mencapai garis finish yang jaraknya sepanjang 8800 meter hanya dalam waktu 55 detik saja, semua penonton bertanya-tanya, berapa sih kecepatan motor si pembalap?

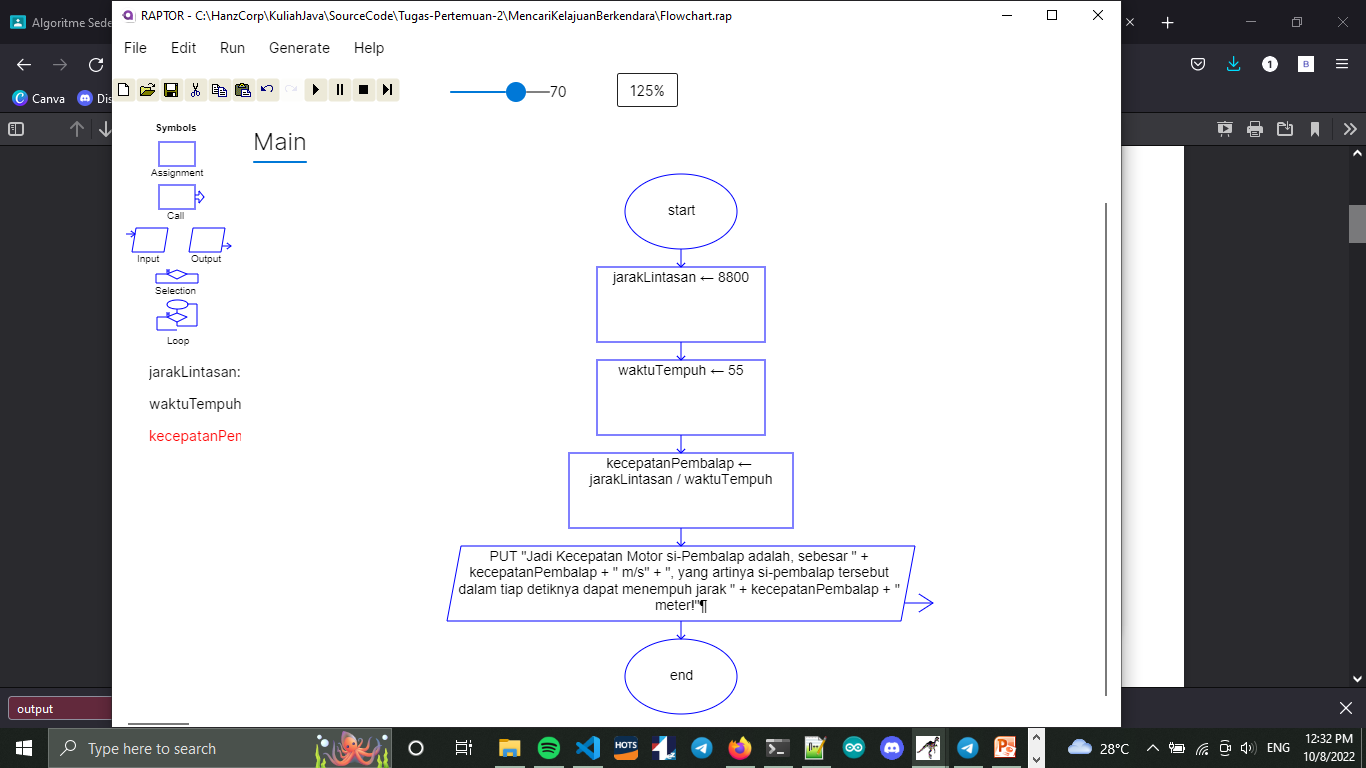
# Algoritma Deskriptif

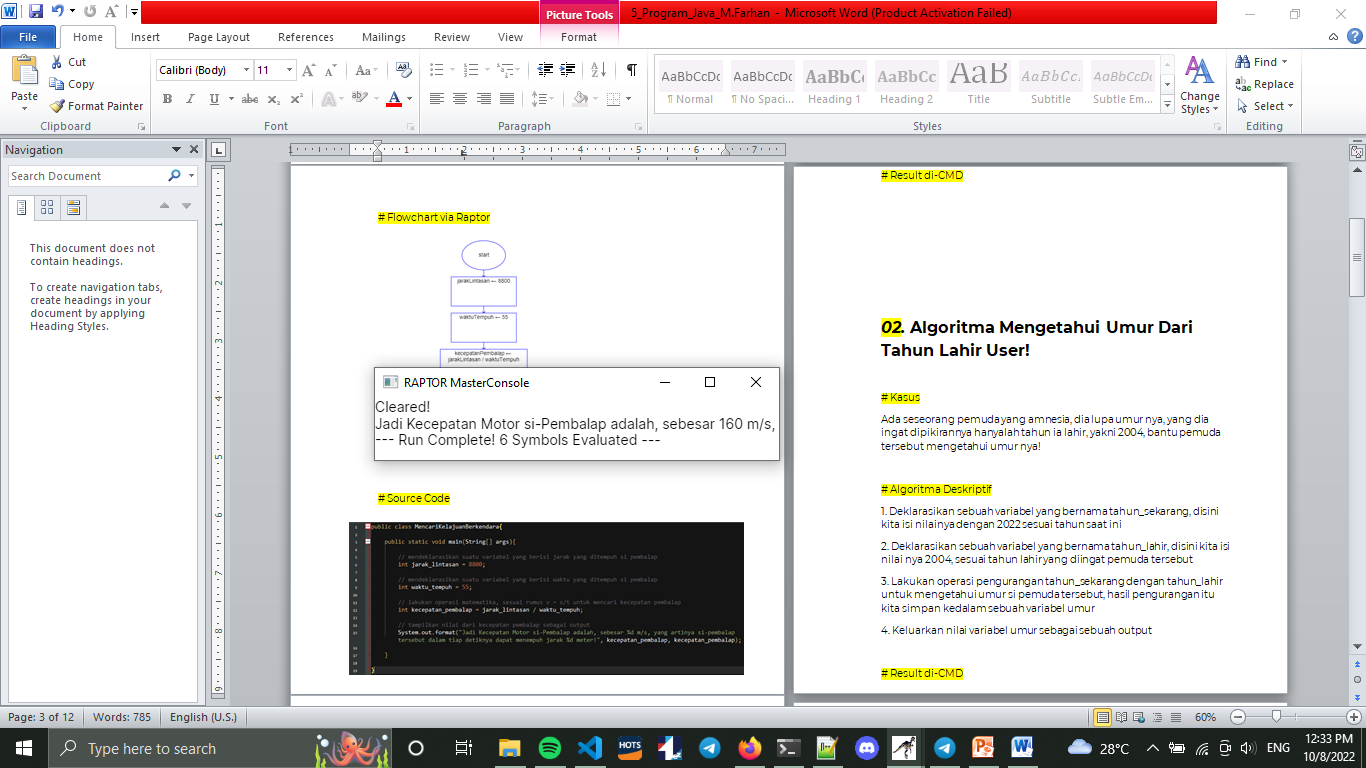
1. Deklarasikan sebuah variabel yang bernama jarakLiintasan, dimana didalamnya berisi sebuah nilai 8800 meter (sesuai nilai jarak yang yang di tempuh pembalap pada kasus diatas)

2. Deklarasikan lagi sebuah variabel yang kita beri nama waktuTempuh, dimana didalamnya berisi sebuah nilai 55 detik (sesuai dengan waktu yang di tempuh pembalap sampai finish pada kasus diatas)

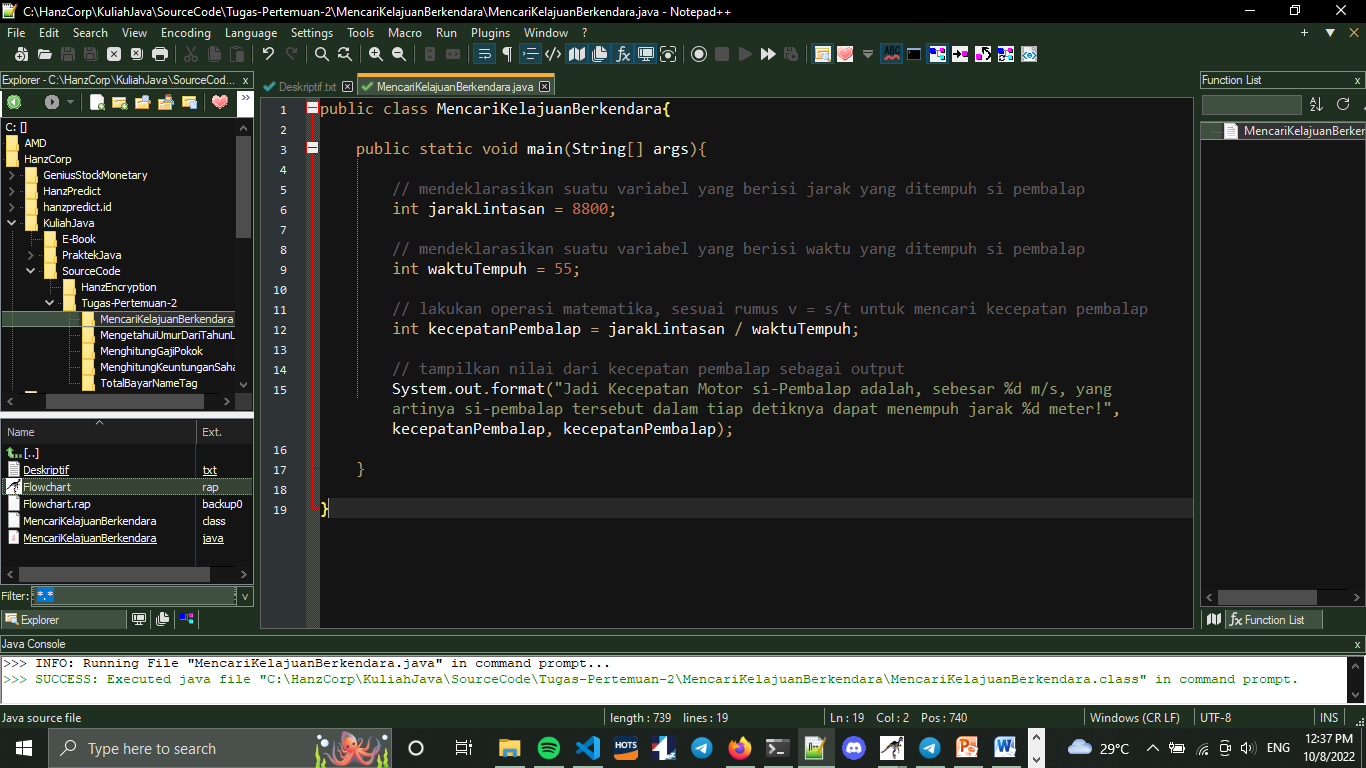
3. Lakukan operasi matematika sesuai rumus kecepatan, yakni untuk mendapatkan nilai kecepatan, lakukan pembagian antara jarakLintasan dan waktuTempuh, dimana nilai dari hasil tersebut akan kita simpan dalam sebuah variabel yang kita namakan kecepatanPembalap

4. Kita akan mengeluarkan/print nilai dari kecepatanPembalap sebagai output akhir

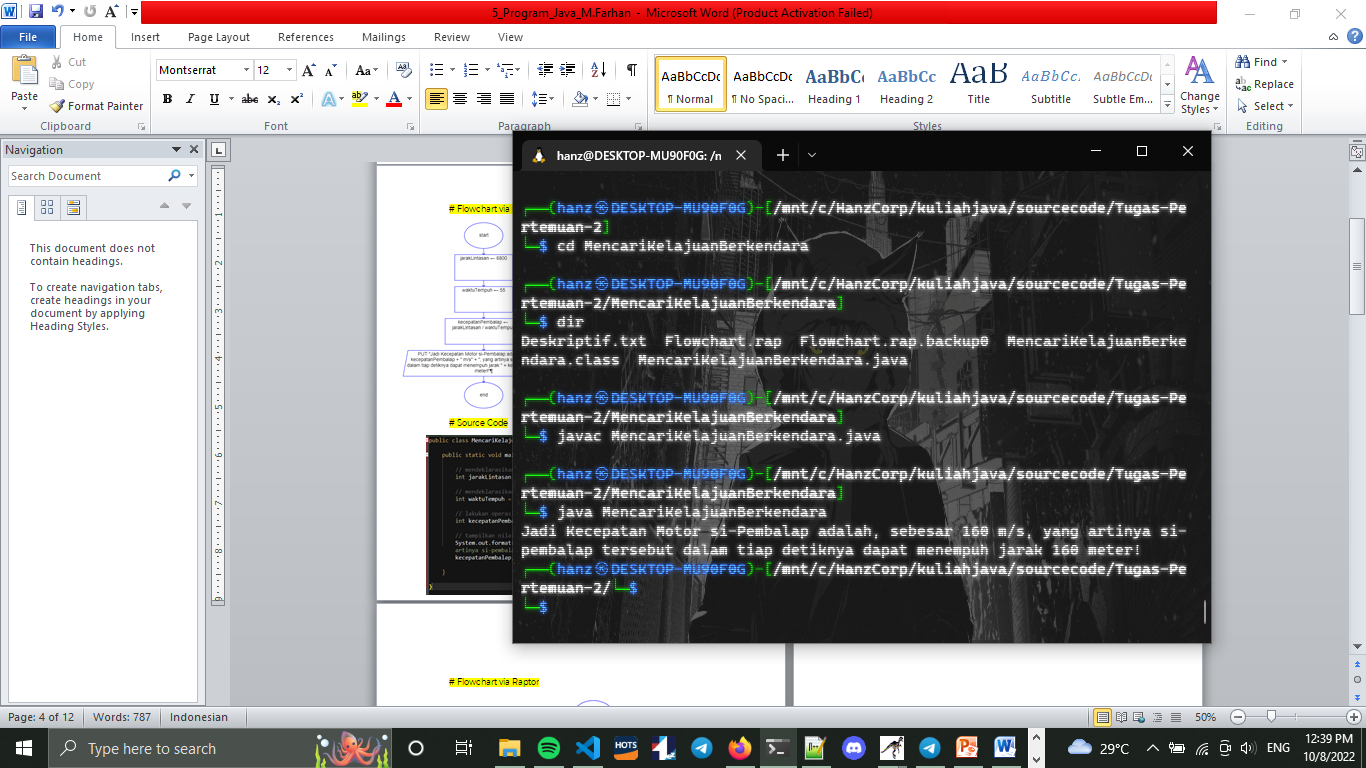
# Flowchart via Raptor beserta Resultnya



# Source Code



# Result di-CMD



***02.* Algoritma Mengetahui Umur Dari Tahun Lahir User!**

# Kasus

Ada seseorang pemuda yang amnesia, dia lupa umur nya, yang dia ingat dipikirannya hanyalah tahun ia lahir, yakni 2004, bantu pemuda tersebut mengetahui umur nya!

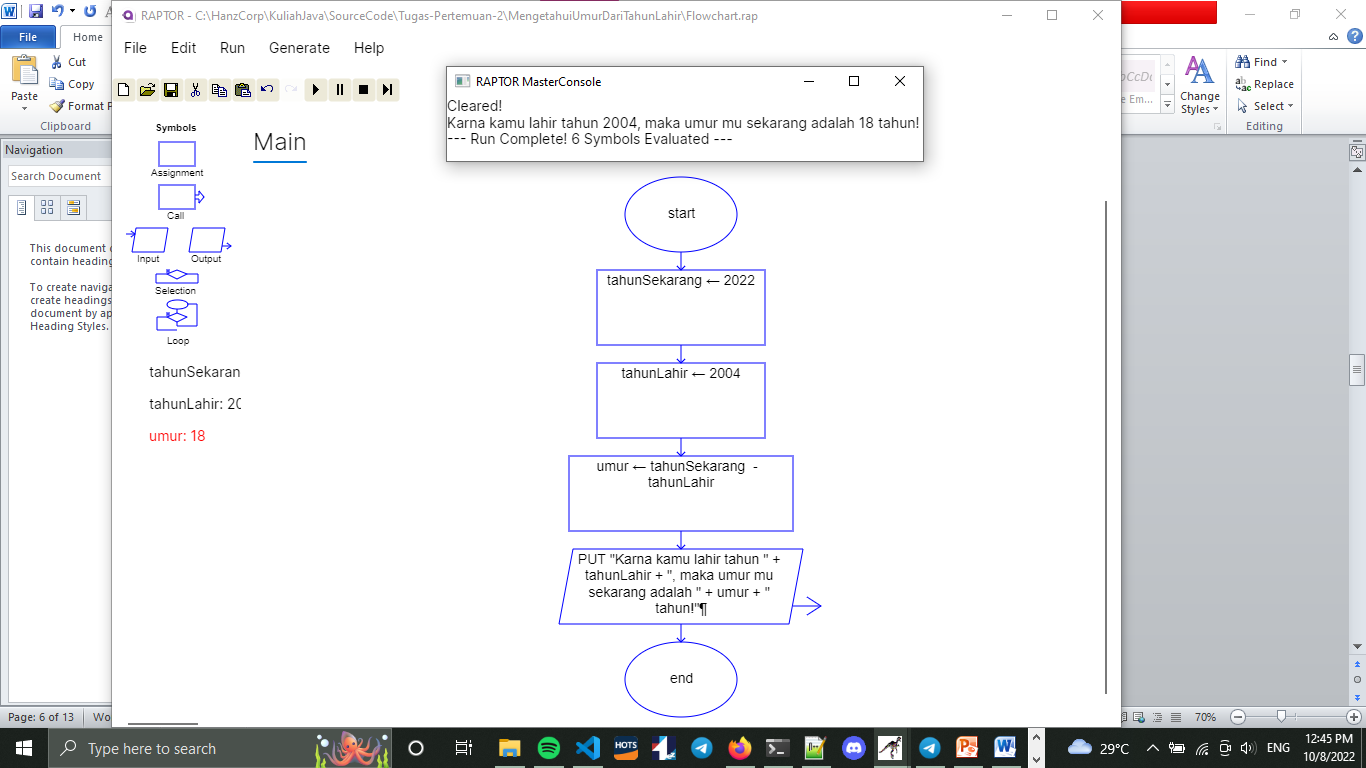
# Algoritma Deskriptif

1. Deklarasikan sebuah variabel yang bernama tahunSekarang, disini kita isi nilainya dengan 2022 sesuai tahun saat ini

2. Deklarasikan sebuah variabel yang bernama tahunLahir, disini kita isi nilai nya 2004, sesuai tahun lahir yang diingat pemuda tersebut

3. Lakukan operasi pengurangan tahunSekarang dengan tahunLahir untuk mengetahui umur si pemuda tersebut, hasil pengurangan itu kita simpan kedalam sebuah variabel umur

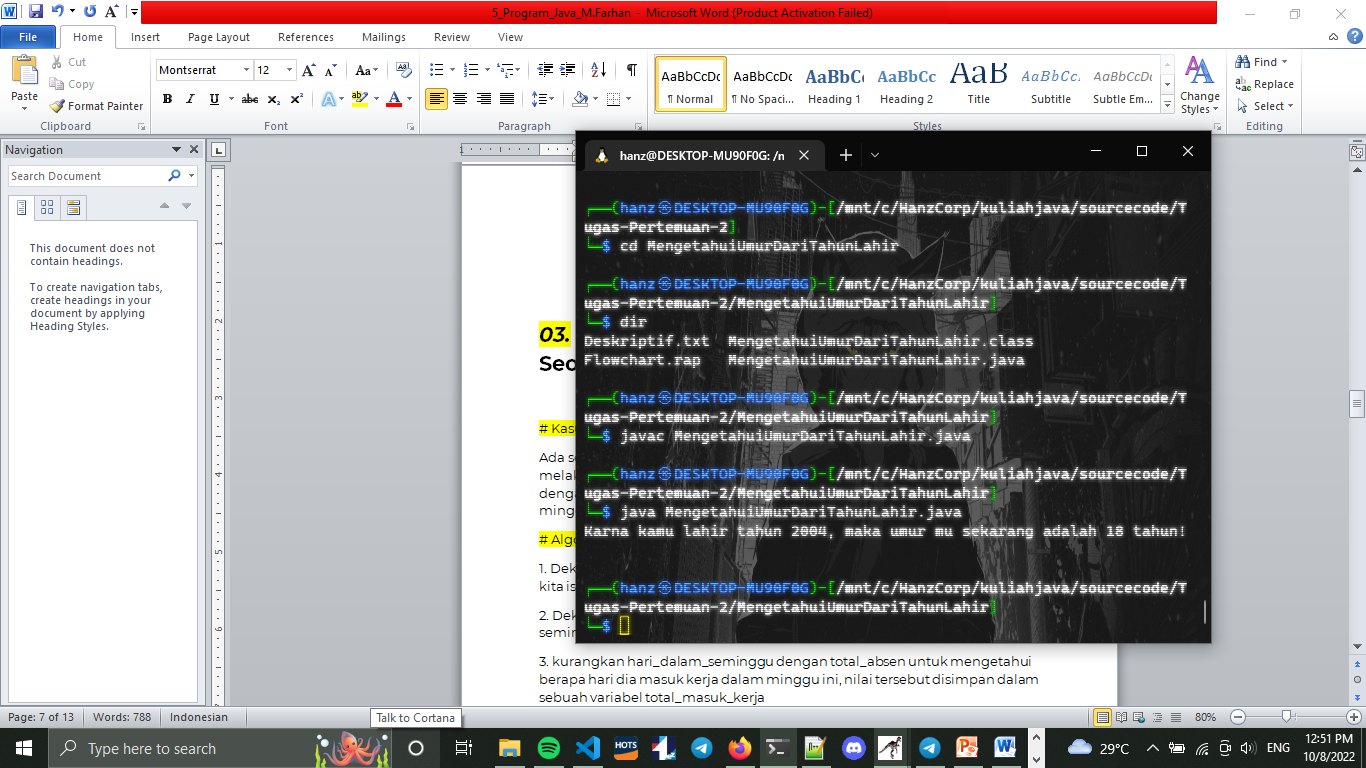
4. Keluarkan nilai variabel umur sebagai sebuah output

# Flowchart via Raptor

# Source Code



# Result Dari CMD



***03.* Algoritma Mengetahui Gaji Pokok Seorang Pekerja Dalam Seminggu!**

# Kasus

Ada seorang buruh yang kantor nya full masuk seminggu, kantornya pun melakukan sistem gaji mingguan, dimana perhitungan gaji nya di sesuaikan dengan berapa kali ia masuk kantor, gaji pokok nya perhari 25 ribu, kebetulan minggu ini dia absen 3 hari, cari tahu berapa gaji dia minggu ini!

# Algoritma Deskriptif

1. Deklarasikan sebuah variabel yang bernama hariDalamSeminggu, disini kita isikan nilai total hari dalam seminggu, yakni 7

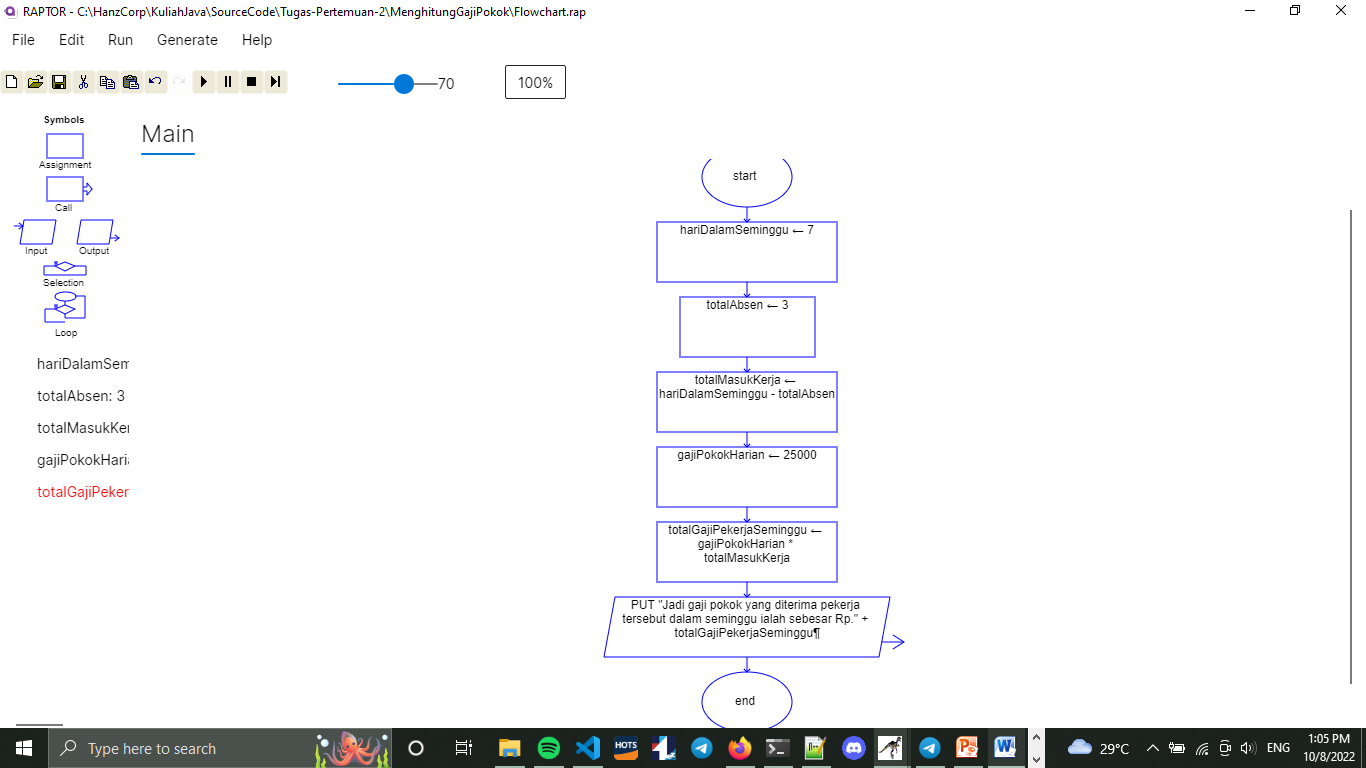
2. Deklarasikan variabel totalAbsen yang memuat total dia absen dalam seminggu, disini ada 3 hari dia absen, maka nilainya sama dengan 3

3. kurangkan hariDalamSeminggu dengan totalAbsen untuk mengetahui berapa hari dia masuk kerja dalam minggu ini, nilai tersebut disimpan dalam sebuah variabel totalMasukKerja

4. Deklarasikan variabel yang memuat gaji pokok harian si pekerja, nilai ini disimpan dalam variabel gajiPokokHarian, dimana disini 25000 nilainya

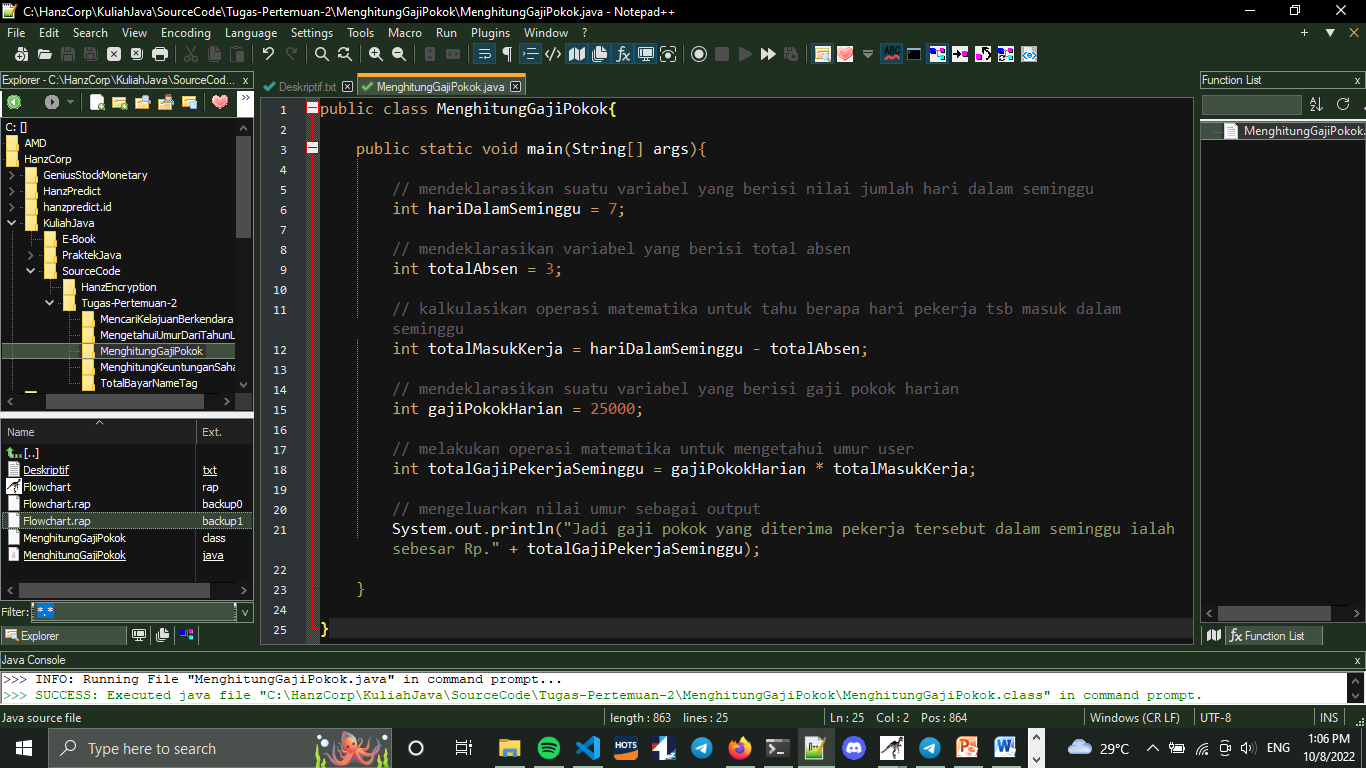
5. Lalu kita kalikan gajiPokokHarian dengan totalMasukKerja untuk mendapatkan hasilnya, nilai dari operasi matematika ini kita simpan dalam variabel totalGajiPekerjaSeminggu

6. Print atau keluarkan nilai variabel totalGajiPekerjaSeminggu sebagai sebuah output

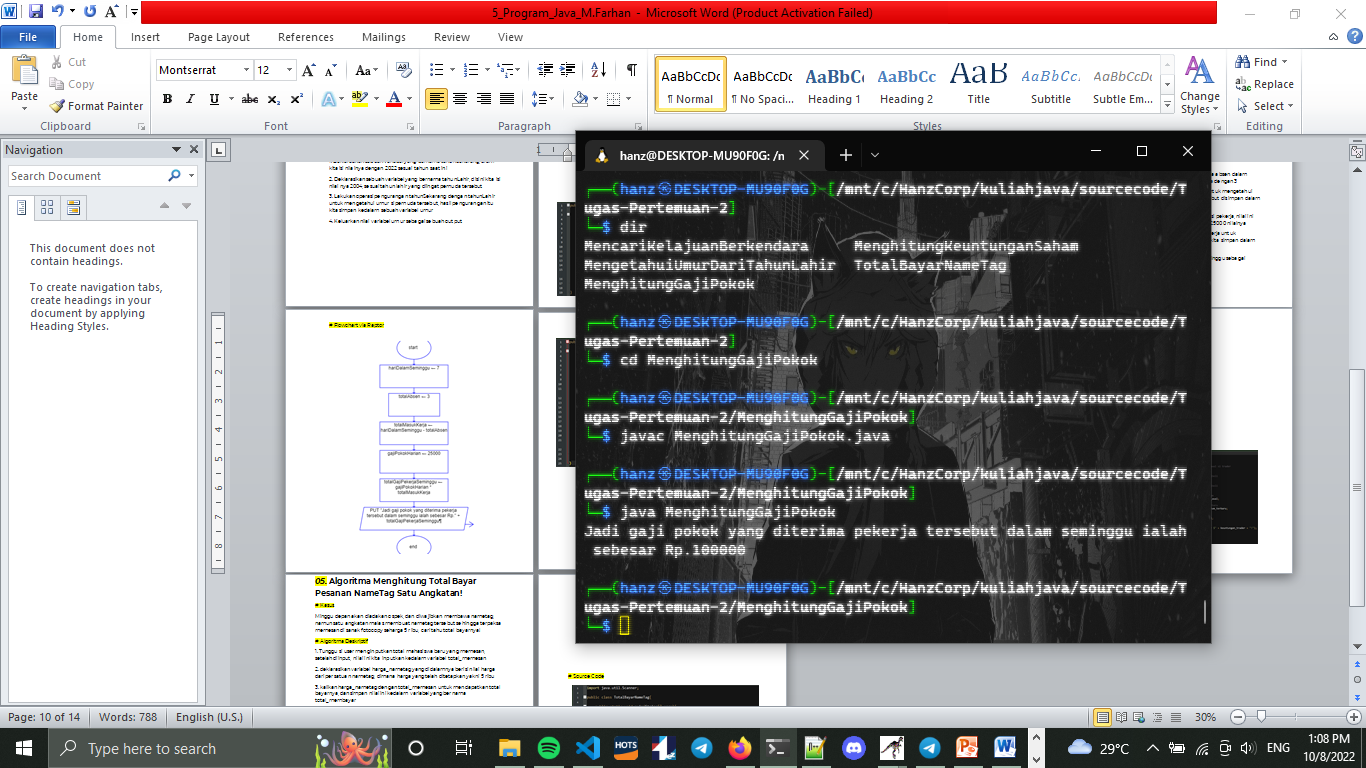
# Flowchart via Raptor



# Source Code



# Result di CMD



***04.* Algoritma Menghitung Keuntungan Jual Beli Saham Seorang Trader!**

# Kasus

Ada seorang trader yang membeli 500 lembar saham Apple seharga $20 per-lembarnya, namun tak lama kemudian harga saham Apple tersebut terbang ke harga $25 per-lembarnya, disini kita akan mencari keuntungan trader tersebut!

# Algoritma Deskriptif

1. Deklarasikan sebuah variabel yang bernama jumlahLembarPembelianAwal yang akan kita isikan sebuah nilai total jumlah pembelian lembar saham awal si trader tersebut, disini kita isikan 500 lembar

2. Deklarasikan sebuah variabel yang bernama hargaSahamAwal yang akan kita isikan sebuah nilai harga awal saham tersebut, disini kita isikan $20

3. Deklarasikan sebuah variabel yang bernama hargaSahamTerbaru yang akan kita isikan sebuah nilai harga terakhir saham tersebut setelah dibeli, disini kita isikan $25

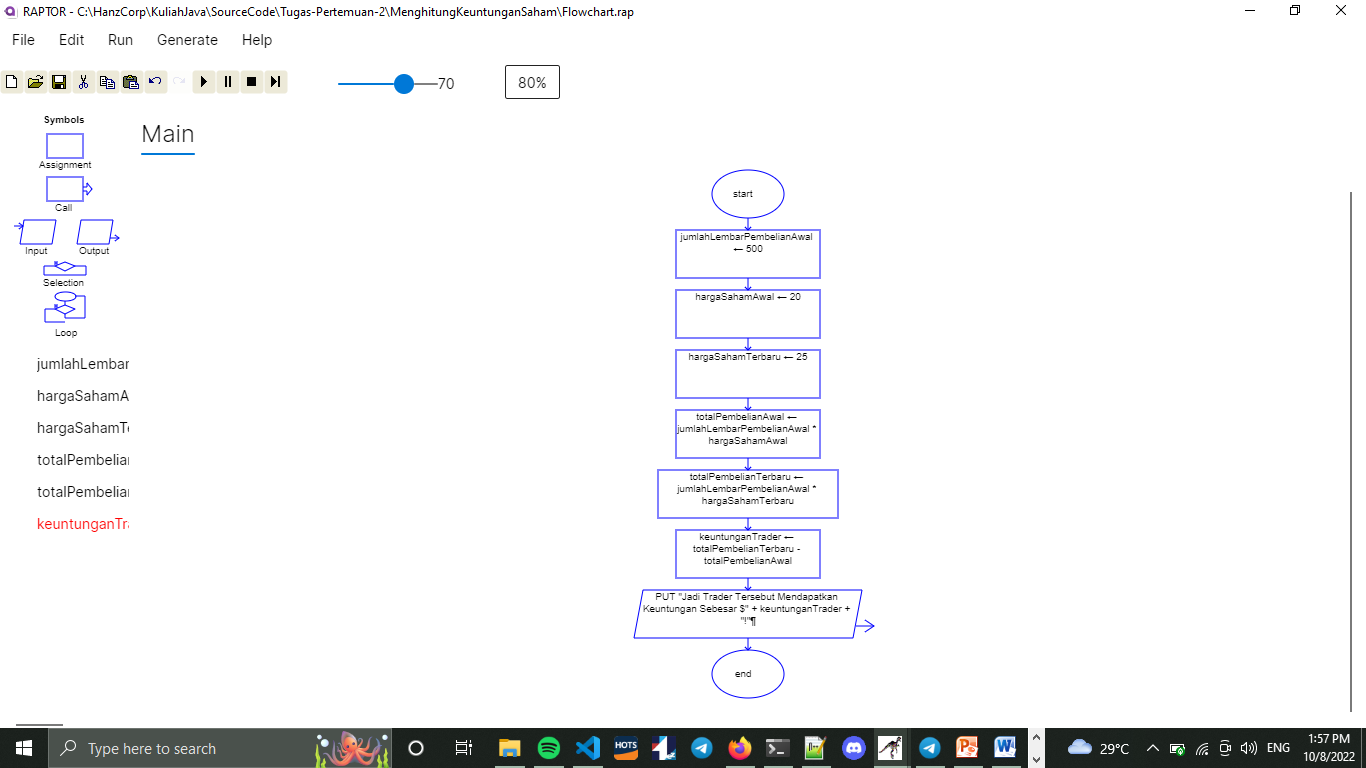
4. Kalikan jumlahLembarPembelianAwal dengan hargaSahamAwal untuk mendapatkan harga total pembelian awal saham, dimana nilai tersebut akan kita simpan kedalam sebuah variabel yang bernama totalPembelianAwal

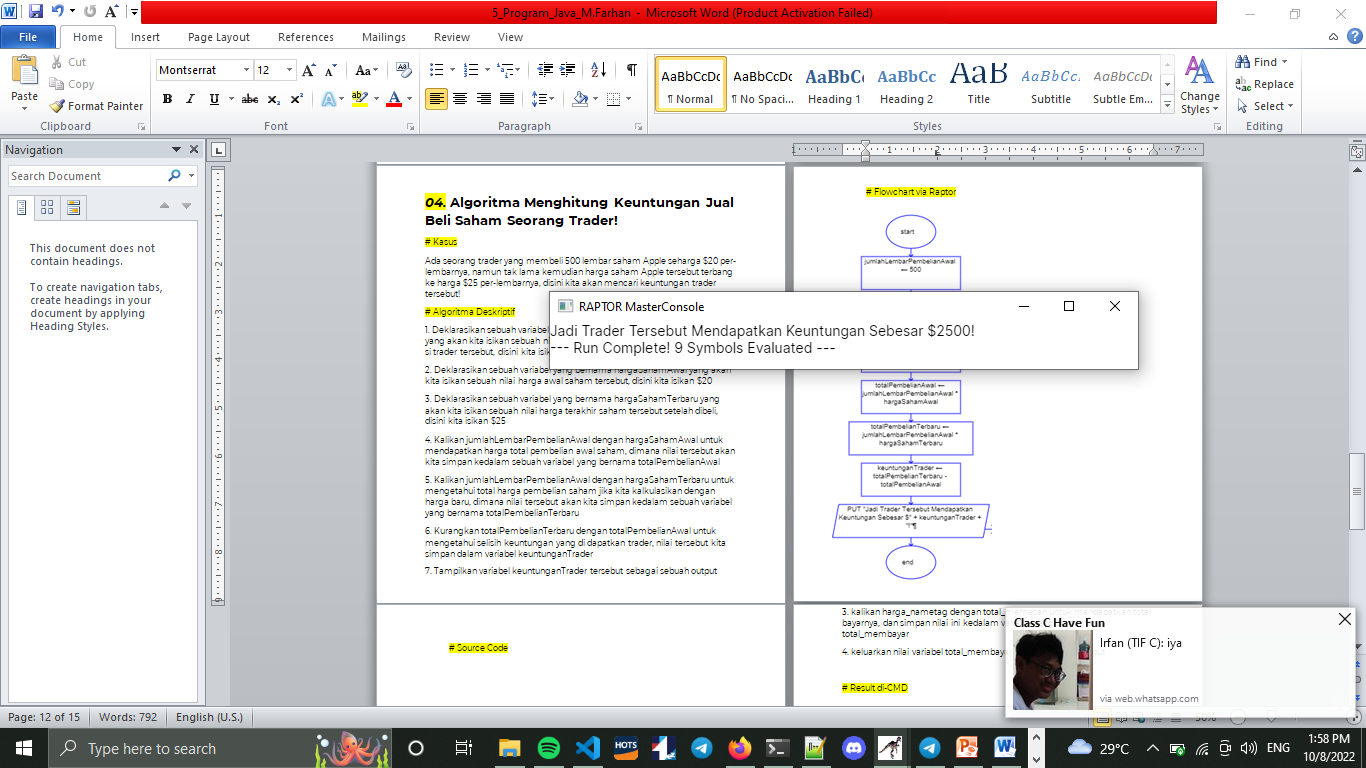
5. Kalikan jumlahLembarPembelianAwal dengan hargaSahamTerbaru untuk mengetahui total harga pembelian saham jika kita kalkulasikan dengan harga baru, dimana nilai tersebut akan kita simpan kedalam sebuah variabel yang bernama totalPembelianTerbaru

6. Kurangkan totalPembelianTerbaru dengan totalPembelianAwal untuk mengetahui selisih keuntungan yang di dapatkan trader, nilai tersebut kita simpan dalam variabel keuntunganTrader

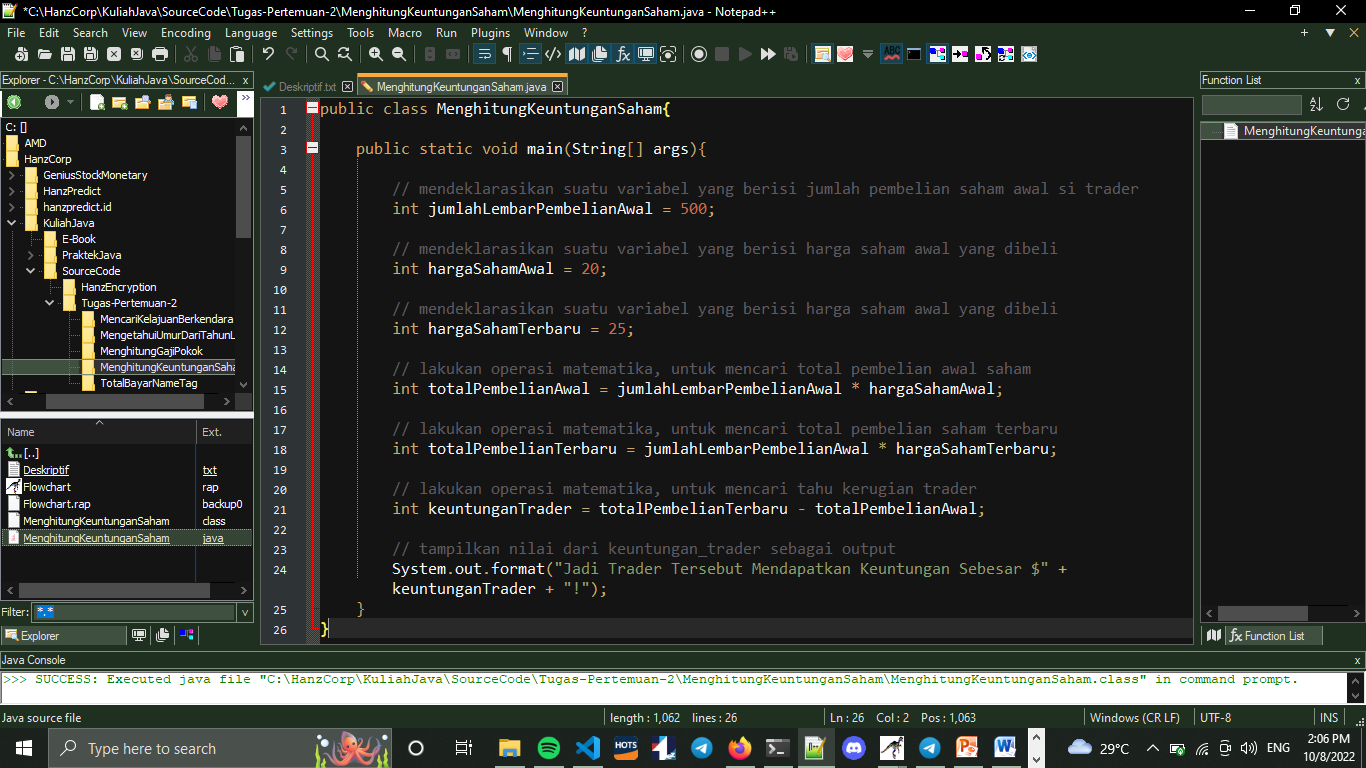
7. Tampilkan variabel keuntunganTrader tersebut sebagai sebuah output

# Flowchart via Raptor

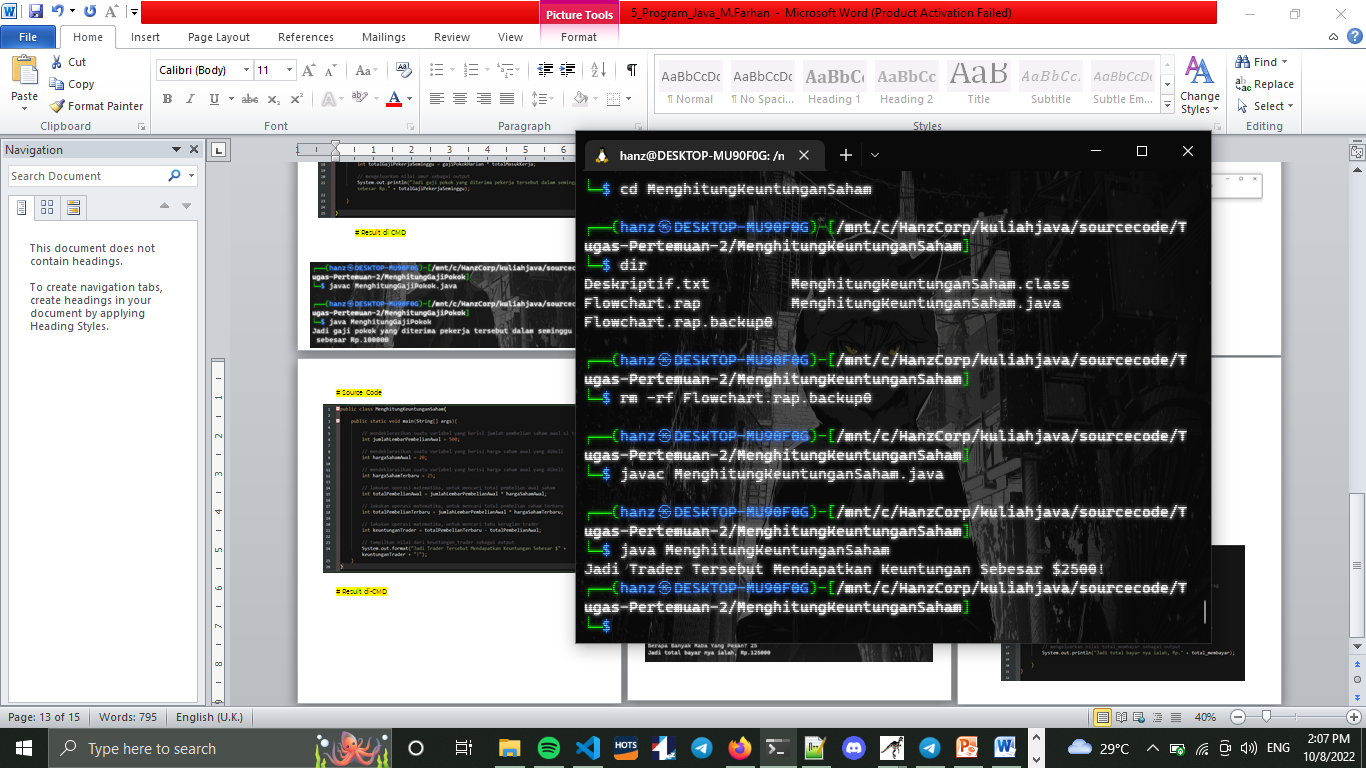




# Source Code



# Result di-CMD



***05.* Algoritma Menghitung Total Bayar Pesanan NameTag Satu Angkatan!**

# Kasus

Minggu depan akan diadakan ospek, dan diwajibkan membawa nametag, namun satu angkatan malas membuat nametag tersebut sehingga terpaksa memesan di sanak fotocopy seharga 5 ribu, cari tahu total bayarnya!

# Algoritma Deskriptif

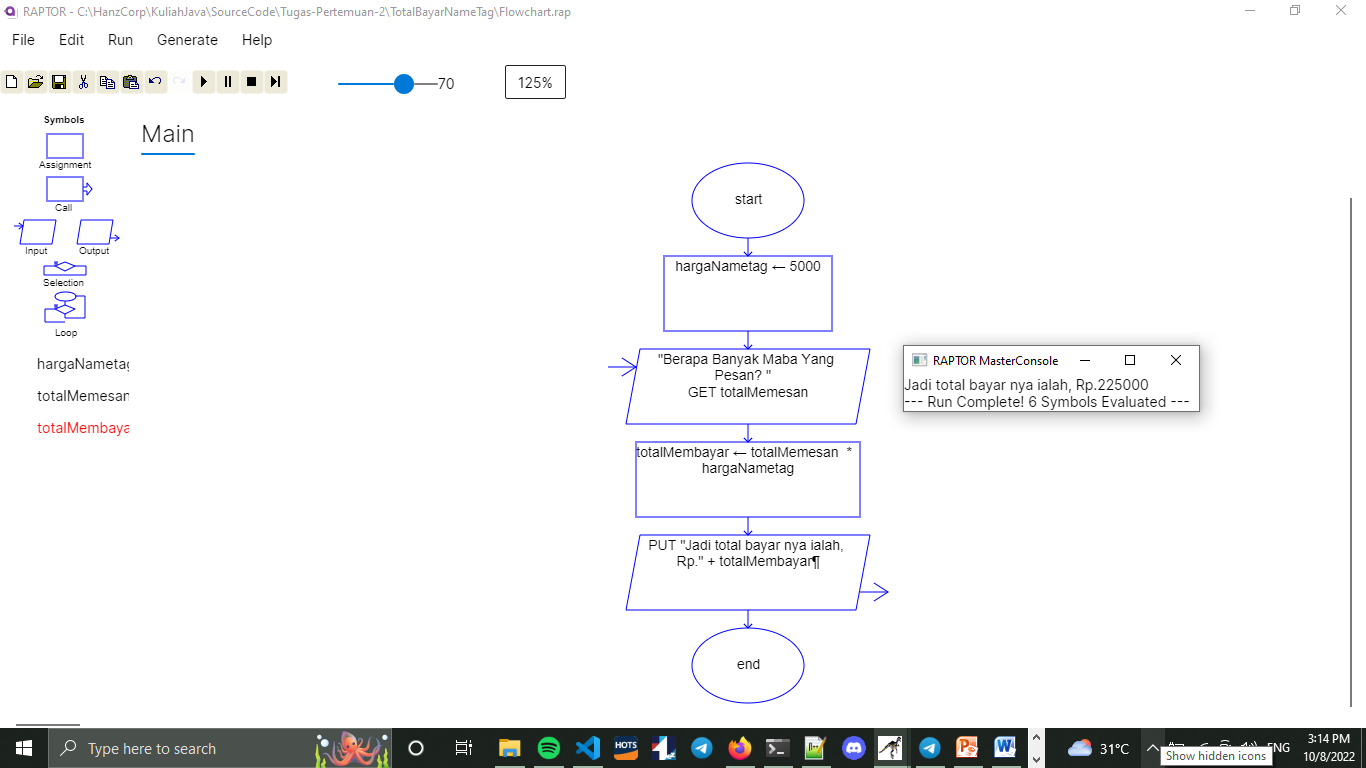
1. Tunggu si user menginputkan total mahasiswa baru yang fix memesan, setelah diinput, nilai ini kita masukkan kedalam variabel totalMemesan

2. deklarasikan variabel hargaNametag yang didalamnya berisi nilai harga dari per satuan nametag, dimana harga yang telah ditetapkan yakni 5 ribu

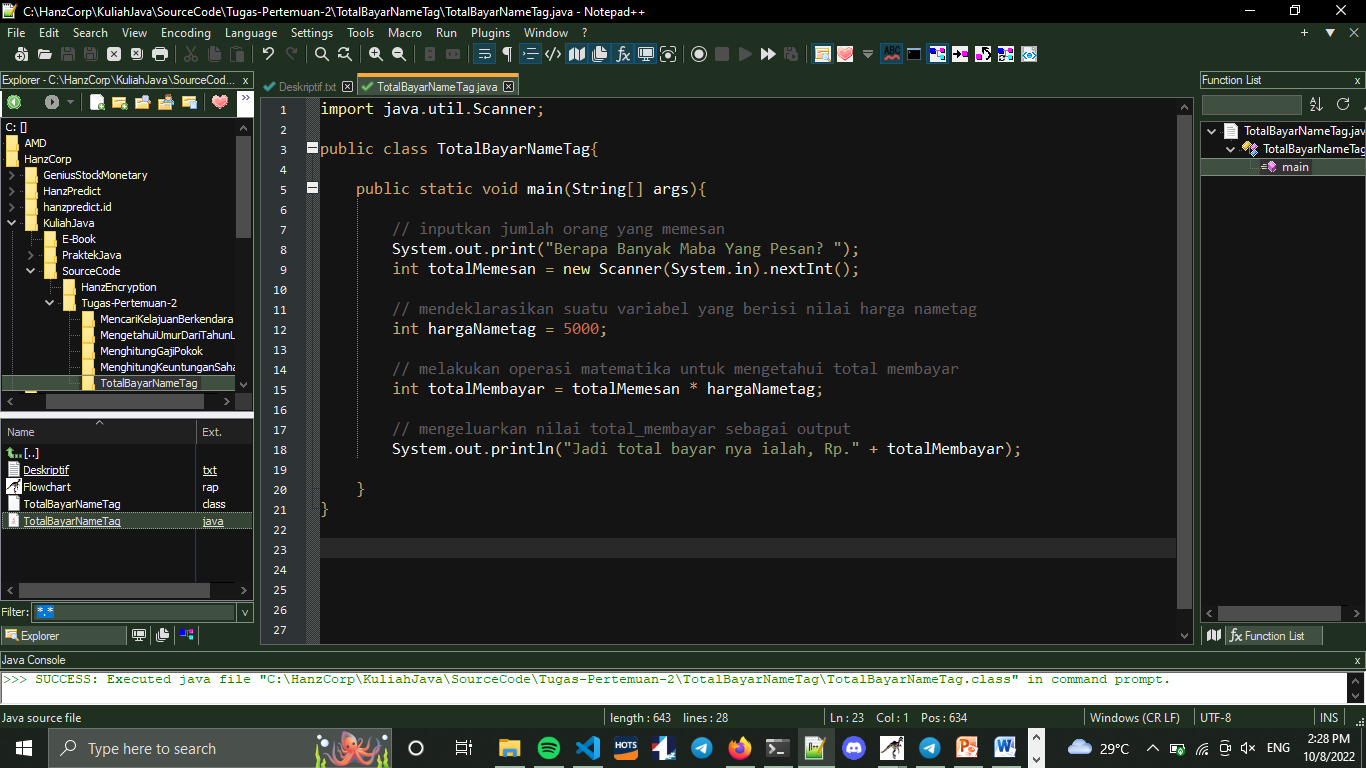
3. kalikan hargaNametag dengan totalMemesan untuk mendapatkan total bayarnya, dan simpan nilai ini kedalam variabel yang bernama totalMembayar

4. keluarkan nilai variabel totalMembayar sebagai sebuah output

# Flowchart via Raptor



# Source Code



# Results di CMD

