



# Project Base Learning

## Menghitung Gaji Pegawai

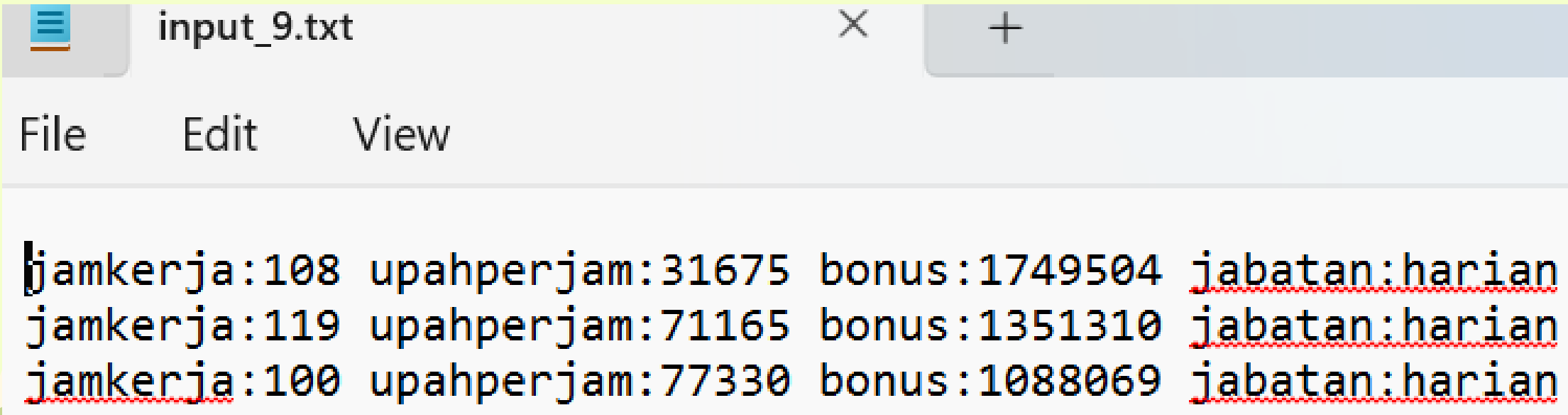
Oleh Auditya Merosya Putri  
NIM 24343001

# Penjelasan Program

Program ini memiliki fitur utama untuk menghitung gaji karyawan dengan menggunakan input dari file teks input.txt. Kemudian program akan memproses setiap data yang ada di dalam file input.txt dan ditampilkan pada file output.txt. Program ini juga menggunakan fitur untuk menghitung waktu yang dibutuhkan oleh program dalam satuan detik.

# Input

- Input program menggunakan file input berupa file teks yang bernama input\_9.txt yang berisi data pegawai dengan format, di mana setiap baris file input memuat data untuk satu pegawai.
- `fopen("input_9.txt", "r")` membuka file dalam mode baca.
- Data dari file diinputkan dengan fungsi `fscanf()`.



```
input_9.txt
File Edit View
jamkerja:108 upahperjam:31675 bonus:1749504 jabatan:harian
jamkerja:119 upahperjam:71165 bonus:1351310 jabatan:harian
jamkerja:100 upahperjam:77330 bonus:1088069 jabatan:harian
```

# Proses Yang Terjadi Pada Input

1. Program akan membuka file input bernama input\_9.txt untuk membaca data yang ada di dalam file tersebut. Fungsi `fopen("input_9.txt", "r")` akan membuka file input\_9.txt dalam mode baca ("r").
2. Jika file tidak ditemukan saat membuka file, maka `fopen()` akan mengembalikan NULL dan jika file gagal dibuka, program mencetak pesan kesalahan dan keluar dengan nilai 1 menggunakan `return 1;`.
3. `malloc(sizeof(struct Pegawai))` mengalokasikan memori sebesar ukuran struktur Pegawai.
4. Fungsi `fscanf()` digunakan untuk membaca data dari file input dan menyimpannya ke dalam struktur pegawai pada indeks jumlahPegawai.

# Proses Yang Terjadi Pada Input

5. Pembacaan data dilakukan dengan perulangan while yang akan terus berjalan selama ada data bisa dibaca (belum mencapai EOF / End Of File). Setiap kali data satu pegawai dibaca, data tersebut disimpan ke dalam array pegawai.

6. Program melakukan perhitungan untuk mendapatkan gaji total. Hasil perhitungan disimpan dalam gajiTotal di struktur pegawai[jumlah pegawai].

7. realloc() untuk mengalokasikan ulang memori array pegawai sehingga dapat menampung data pegawai berikutnya atau menambah ukuran array agar bisa menyimpan data pegawai tambahan baik dalam jumlah besar maupun kecil.



# Output

1. Program akan membuka file output.txt dengan `fopen("output_9.txt", "w")` membuka file output dengan mode tulis ("w"). Jika file gagal dibuka, program mencetak pesan kesalahan dan keluar.
2. Program menampilkan "Data Gaji Pegawai" menggunakan `fprintf()` setiap baris perbaris. Dengan perulangan while, program menulis data masing-masing pegawai dengan format seperti disamping.



```
File Edit View
Data Gaji Pegawai
Jabatan: harian, Jam Kerja: 108, Upah/Jam: 31675.00, Bonus: 1749504.00, Gaji Total: 5170404.00
-----
Jabatan: harian, Jam Kerja: 119, Upah/Jam: 71165.00, Bonus: 1351310.00, Gaji Total: 9819945.00
```

# Output

4. Output dari waktu eksekusinya.

Dimana program nantinya akan mencetak waktu eksekusi yang akan ditampilkan dilayar seperti berikut :

5. Setelah semua program selesai, Program mencetak pesan hasil bahwa data telah diproses dan disimpan di file output\_9.txt.

```
-jzuepepo.Ply --student-MICROSOF C-MIEngine-ERROR-xm1svv  
Data telah diproses dan disimpan di file output_9.txt  
Waktu eksekusi program: 0.02 detik  
PS C:\Users\USER> |
```

# Waktu Eksekusi

clock\_t fungsi library time.h yang digunakan untuk mencatat waktu program.

- clock\_t start = clock(); mencatat waktu mulai program berjalan.
- clock\_t end = clock(); digunakan mencatat waktu setelah seluruh proses selesai (pembacaan, perhitungan, penulisan)
- Waktu eksekusi dihitung dengan selisih antara waktu akhir dan waktu mulai, dibagi dengan CLOCKS\_PER\_SEC untuk mendapatkan hasil dalam satuan detik ,seperti berikut:

```
//Hitung waktu eksekusi dalam detik  
double waktuEksekusi = (double)(end - start) / CLOCKS_PER_SEC;
```





**Terima kasih**