**Inklusi Library**:

#include <iostream>

#include <string>

using namespace std;

Program ini menggunakan **iostream** untuk input dan output, serta **string** untuk memproses teks. **using namespace std;** digunakan agar kita tidak perlu menulis **std::** setiap kali menggunakan elemen dari namespace **std**.

**Deklarasi Variabel**:

string **namaKaryawan**;

int **golongan, jamKerja**;

**namaKaryawan** menyimpan nama karyawan, **golongan** menyimpan golongan kerja (2 atau 3), dan **jamKerja** menyimpan jumlah jam kerja dalam seminggu.

**Input Data Karyawan**:

cout << "Masukkan nama karyawan: ";

**getline**(cin, namaKaryawan);

cout << "Masukkan golongan (2 atau 3): ";

cin >> golongan;

cout << "Masukkan jumlah jam kerja dalam seminggu: ";

cin >> jamKerja;

Program meminta pengguna memasukkan nama karyawan, golongan, dan jumlah jam kerja. **getline** digunakan untuk membaca nama karyawan yang mungkin terdiri dari beberapa kata.

**Menghitung Upah Pokok Berdasarkan Golongan**:

int upahPokok;

if (golongan == 2) {

upahPokok = jamKerja \* 35000;

}

else if (golongan == 3) {

upahPokok = jamKerja \* 50000;

}

else {

cout << "Golongan tidak valid. Harap masukkan 2 atau 3." << endl;

return 1;

}

Program menghitung upah pokok berdasarkan **golongan**. Golongan 2 memiliki upah per jam Rp 35.000, dan golongan 3 memiliki upah per jam Rp 50.000. Jika golongan tidak valid (bukan 2 atau 3), program akan menampilkan pesan kesalahan dan keluar.

**Menghitung Uang Lembur**:

int uangLembur = 0;

if (jamKerja > 48) {

uangLembur = (jamKerja - 48) \* 10000;

}

Jika jumlah jam kerja melebihi 48 jam, maka jam kerja di atas 48 jam akan dihitung sebagai lembur dengan upah lembur sebesar Rp 10.000 per jam.

**Menghitung Gaji Total**:

int gajiTotal = upahPokok + uangLembur;

Program menjumlahkan **upahPokok** dan **uangLembur** untuk mendapatkan **gajiTotal**.

**Menampilkan Hasil**:

cout << "Gaji Total: Rp " << gajiTotal << endl;

Program menampilkan gaji total karyawan dalam format rupiah.

**Akhir Program**:

return 0;

Program selesai dengan mengembalikan nilai **0**, yang menandakan bahwa program berakhir tanpa kesalahan.