Tugas Proyek Makassar, 20 Juni 2020

**LAPORAN TUGAS PROYEK**

**”PROGRAM MENGHITUNG GAJI PEGAWAI”**

****

B

|  |  |
| --- | --- |
| Nama | : Adrya Dwiprawira Munaf |
| Stambuk | : 13020190088 |
| Kelas | : A2 |
|  |  |
|  |  |

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS MUSLIM INDONESIA**

**MAKASSAR**

**2019**

**PENDAHULUAN**

pada awal tahun 1970-an seorang bernama Bjarne Stroustrup mulai mengembangkan bahasa pemrograman c++ yang merupakan turunan dari Bahasa C.

Jika Bahasa C merupakan bahasa prosedural maka Bahasa C++ hadir menggunakan konsep pemrograman berorientasi objek atau Object Oriented Programming.

Pemrograman dengan bahasa C menggunakan 2 konsep:

* **Data** - bahan yang akan diolah.
* **Algorithms** - metode pengolahan.

Tapi bagaimanapun juga, penggunaan bahasa prosedural seperti C sangat tidak efisien khususnya untuk pembuatan software berskala menengah ke atas atau project besar.

Terlalu banyak perulangan, percabangan, dan kode yang harus kita ketikan.

Maka dari itu kemudian bahasa c ini dikembangkan ke tingkat lebih lanjut menjadi bahasa c yang berorientasi object yang kemudian disebut dengan bahasa C++.

Berikut studi kasus perhitungan gaji pegawai yang akan di atasi dengan Bahasa pemrograman C++:

a. Pengguna memasukkan jumlah pegawai kemudian mengisi data pegawai: nip(nomor induk pegawai), nama pegawai, gaji pokok, golongan dan jam kerja, lembur dimasukkan melalui keyboard

b. Gaji bersih pegawai dihitung dari gaji pokok dan tunjangan dipotong pajak yang harus dibayar. Besar tunjangan adalah 15 % dari gaji pokok, dan besar pajak adalah 10 % dari gaji pokok ditambah tunjangan

c. Gaji pegawai ditentukan berdasarkan gaji tetap dan juga dari lamanya bekerja. Gaji tetap dan gaji per jam tergantung kepada golongan pegawai sesuai:

Golongan Gaji tetap (Rp) Upah lembur per jam (Rp)

1 500.000 5000

2 300.000 3000

3 250.000 2000

4 100.000 1500

5 50.000 1000

d. Menghitung total gaji pegawai yang diterima dengan rumus: gaji sesuai golongan + (banyaknya jam lembur \* tarif lembur sesuai golongan karyawan).

e. Ketentuan yang tidak disebutkan di atas dapat anda tentukan sendiri dalam laporan, minimal mengerjakan sesuai dengan kasus.

f. mengurutkan dan mencari data nilai yang diinginkan sesuai kategori

**ANALISIS KASUS**

Berdasarkan hasil analisis penulis terhadap kasus diatas, langkah penyelesaian program untuk kasus diatas membutuhkan setidaknya Procedural fungsi, 1 fungsi utama eksekusi program loops serta sorting dan searching algorithm.

ALGORITMA DENGAN NATURAL LANGUAGE

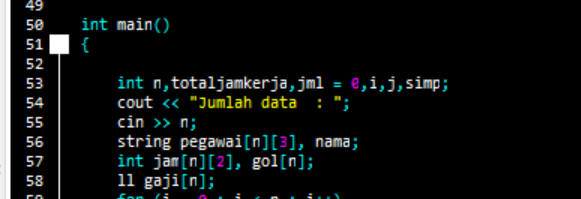
Kasus :

menghitung Gaji bersih pegawai

* Mulai
* Deklarasi tipe data variable n, totaljamkerja, count = 0 , i,j dan simp;
* Menampilkan “Jumlah data: ”
* Menginput n jumlah data pegawai
* Memulai perulangan untuk menginput data pegawai sesuai dengan *n*
* Menginput data pegawai, gaji, jam kerja, dan golongan.
* Memasukkan fungsi gajibersih dengan parameter tunjangan dan gajipokok
* Fungsi gaji bersih mengembalikan nilai gaji pokok ditambah tunjangan
* Membuat fungsi tunjanganPajak dengan parameter gajipokok
* Fungsi tunjanganPajak menghitung tunjangan dengan rumus 15% dari gaji pokok
* Fungsi tunjanganPajak menghitung pajak dengan rumus 10% dari total gaji pokok ditambah tunjangan
* Fungsi tunjanganPajak menampilkan output pajak dan gajibersih dengan memasukkan fungsi panggilan gejibersih()
* Membuat fungsi gajitetap dengan parameter lembur dan golongan
* Membuat conditional statement pada parameter lembur
* Jika jam lembur kurang atau sama dengan nol, maka lembur sama dengan nol, jika tidak maka lembur bernilai sama dengan nilai awal lembur
* Membuat conditional switch case untuk menentukan gaji tetap dengan ketentuan gaji tetap sesuai dengan golongan ditambah jumlah jam lembur dikalikan dengan bonus sesuai dengan golongan.
* Conditional switch case mengambil parameter golongan dan mengembalikan nilai gajitetap.
* Menampilkan output gaji tetap
* Membuat perulangan untuk menampilkan data pegawai yang telah diinput
* Memasukkan fungsi kedalam loops untuk menghitung gaji tetap dan gaji bersih masing masing pegawai
* Membuat loops pengurutan (Sorting) gaji secara ascend
* Menggunakan Bubble sort pada loops untuk mengurutkan gaji secara ascend
* Menampilkan urutan gaji
* Menampilkan tampilan memasukkan nama pegawai
* Menginput nama pegawai yang ingin di search datanya
* Membuat loops pengecekan nama
* Jika nama yang diinput sesuai dengan data pegawai maka tampilkan data pegawai
* Selesai

**HASIL IMPLEMENTASI PROGRAM**

**Deklarasi variable**



Berikut adalah variable dan detail kegunaan dari tipe data Integer:

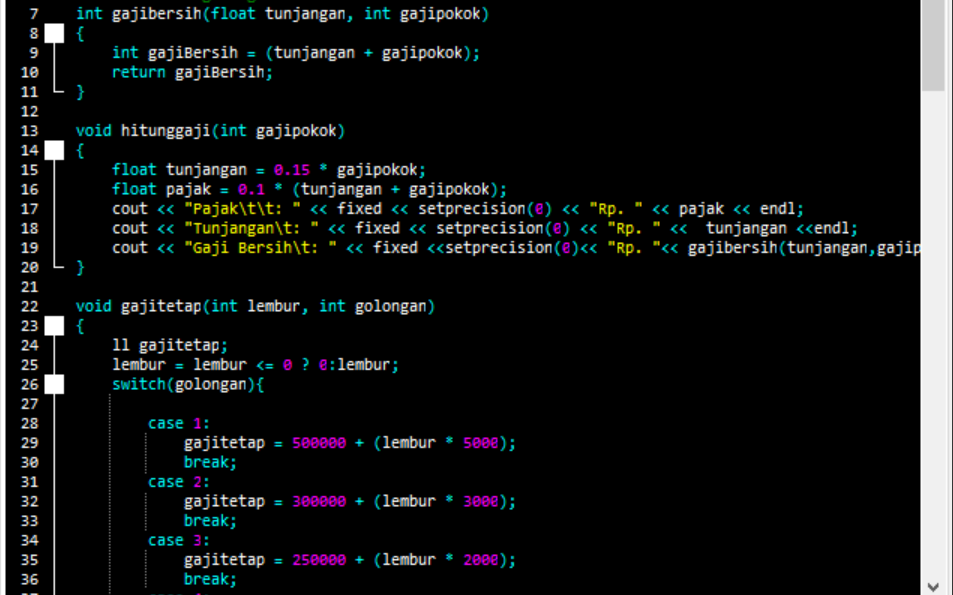
1. *n :* Sebagai acuan inputan jumlah data pegawai, batas dari loops input data pegawai
2. *totaljamkerja* : Sebagai hasil kurang dari total jam kerja dengan jam kerja wajib dan menghasilkan total jam lembur yang akan diproses di fungsi gaji tetap untuk menentukan bonus.
3. *jml = 0* : sebagai variable utama untuk hasil output jumlah data yang ditemukan pada Searching data process.
4. simp : variable bantu untuk digunakan saat proses sorting
5. i dan j : sebagai variable untuk loops

Untuk tipe data string:

1. pegawai[][] : menggunakan array dua dimensi sebagai input data pegawai berupa nama dan nip.

Untuk tipe data integer dan long long array:

1. jam: input jam kerja total dan jam kerja wajib
2. Gaji : input gaji
3. Golongan : input golongan
4. **Pembuatan Fungsi**



1. Fungsi gajibersih

Fungsi gaji bersih akan mengembalikan nilai tambah dari tunjangan dan gajipokok

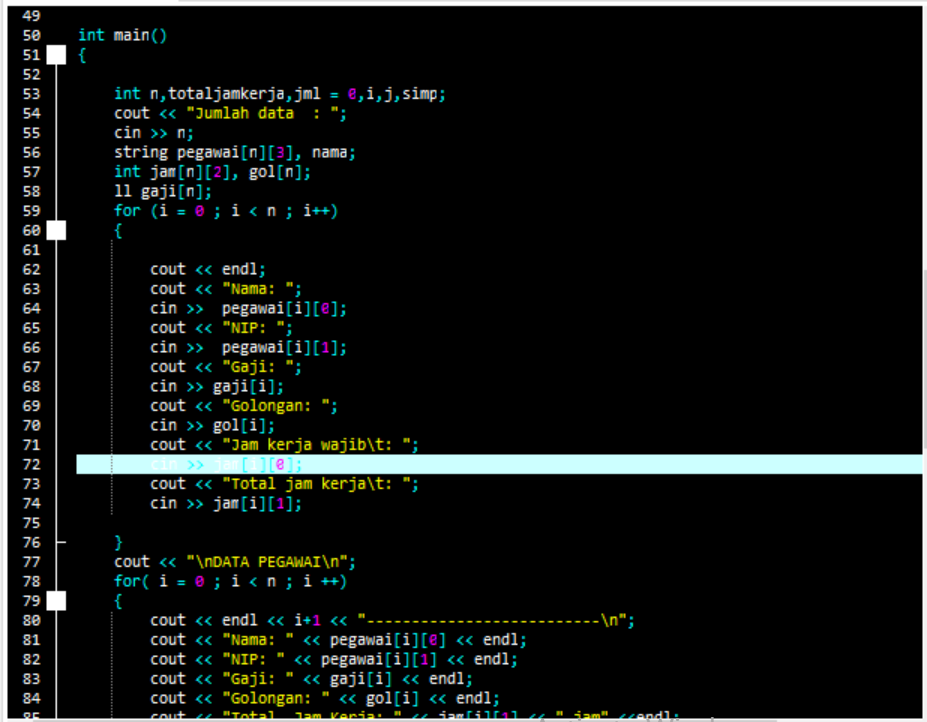
1. Fungsi hitunggaji

Fungsi ini akan memproses jmlah pajak dan tunjangan, selanjutnya fungsi gajibersih akan dipanggil dan akan mengembalikan total gaji yang diterima dikurangi pajak 10%

1. Fungsi gajitetap

Fungsi gaji tetap juga bersifat procedural, fungsi ini akan mengambil parameter golongan dan didalamnya terdapat conditional switch case yang akan memproses bonus dan gaj tetap sesuai dengan golongan dan lama lembur masing-masing.

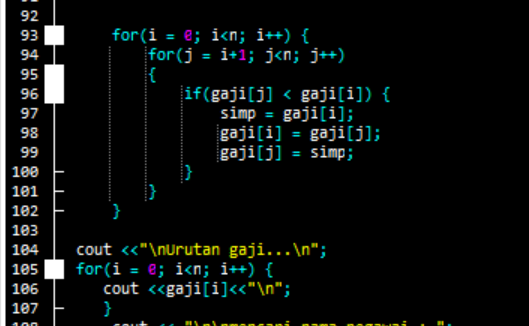
1. **Fungsi Utama int main()**



Fungsi utama akan bertugas mengesekusi semua fungsi yang telah dibuat sebelumnya sesuai dengan inputan dari variable yang tersedia.

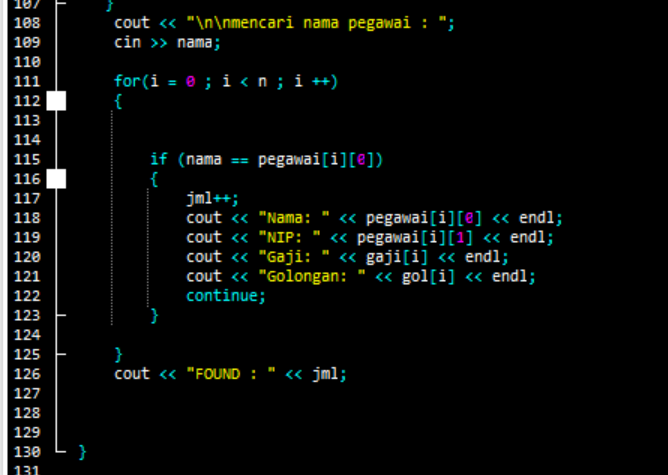
Pada loops pertama, kita dapat menginput data-data pegawai. Pada loops selanjutnya, data pegawai sekaligus detail gaji bersih dan gaji tetap akan ditampilkan.

1. **Sorting**



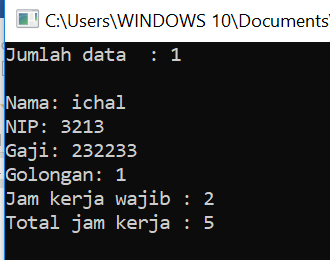
Sorting diatas menggunakan metode bubble sort dan akan mengurutkan gaji pegawai dari yang terendah ke tertinggi

1. **Searching**

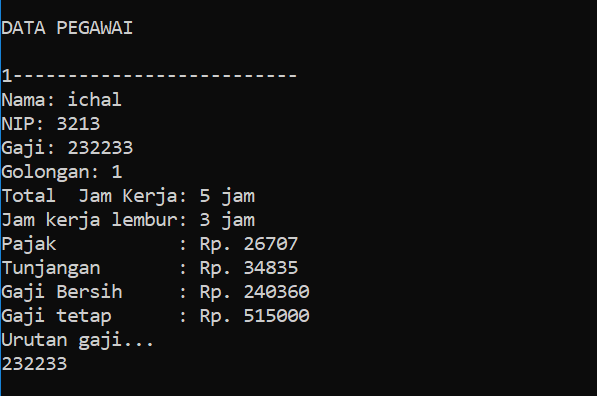


Teknik searching yang digunakan adalah Linear Search karena dikenal dengan Teknik nya yang tidak rumit. Teknik ini hanya menggunakan satu loops dan didalamnya akan terdapat conditional if statements sebagai tolak ukur pengecekan nama yang telah diinput untuk ditampilkan jika ditemukan, dan count akan terincrement untuk menampilkan detail data pegawai yang ditemukan. Jika tidak ditemukan maka jml akan bernilai 0. Berarti data pegawai yang ditemukan tidak ada.

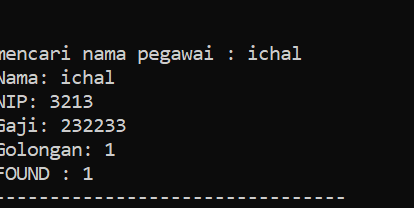
**PENGUJIAN PROGRAM**



Proses penginputan data pegawai sesuai dengan jumlah n data pegawai



Proses selanjutnya adalah implementasi dari fungsi yang akan memproses penghitungan gaji . Dan pada bagian bawah terdapat implementasi dari bubble sort. Karena data yang diinput hanya 1, maka data yang ditampilkan juga 1.



Proses ini adalah implementasi dari linear search yang akan melakukan loops pengecekan pada pegawa[][] jika ditemukan, maka akan ditampilkan data dari pegawai yang bersangkutan.

**PENUTUP**

1. **Kesimpulan**

Agar program dapat berjalan sesuai dengan yang diinginkan , program harus disusun dengan algoritma yang baik dan dapat dipastikan bahwa semua kasus atau permasalah dapat diselesaikan dengan algoritma tersebut

1. **Saran**

Sebaiknya, algoritma yang digunakan lebih diefesienkan karena program yang memakan terlalu banyak baris akan mempengaruhi kecepatan ekeskusi program.

**Daftar Pustaka**

**https://kodedasar.com/belajar-cpp/**

<https://www.softwaretestinghelp.com/sorting-techniques-in-cpp/>

https://www.softwaretestinghelp.com/searching-algorithms-in-cpp/