**LAPORAN PROJECT MEMBUAT PROGRAM UNTUK MENJUMLAHKAN BILANGAN SATU SAMPAI N**

****

**Disusun Oleh:**

**Muhamad Hilmi – 2C1230005**

**Maudina Rachmawati – 2C1230010**

**PROGRAM STUDI AGRIBISNIS FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS KOPERASI INDONESIA**

**KATA PENGANTAR**

Puji Syukur kepada Allah SWT karena lapotan project membuat program penjumlahan bilangan satu sampai n ini dapat terselesaikan. Laporan ini dibuat untuk memenuhi tugas akhir semester satu mata kuliah algoritma, dosen pengajar bapak Agus Nugraha, S.T., M.Kom.

Laporan ini membahas tentang dasar-dasar pemrograman dengan menggunakan bahasa pemrograman C++. Mulai dari pengantar algoritma, dasar-dasar pemrograman, sampai kepada penerapannya dengan menggunakan variasi bentuk perintah yang ada. Tentu saja dengan segala keterbatasan yang ada, lapora ini jauh dari sempurna. Untuk itu masukan dan saran untuk perkembangan modul ini sangat diharapkan. Akhirnya semoga modul ini berguna dalam membantu mahasiswa untuk mempelajari tentang Algoritma dan Pemrograman.

Semoga Laporan ini bermanfaat dan berharga kepada para pembaca. Terima kasih atas dukungan dan kerja sama yang telah diberikan.

Bandung, 24 Januari 2024

Penyusun.

***ABSTRAK***

*Programming is the process of creating computer software by applying certain algorithms and data structures using programming languages. An algorithm is a systematic method and stage used to solve a problem. The data structure is the place where the data storage needs to be organized by a program on a computer. A deep understanding of algorithms is becoming crucial in the development of modern software and computing, ensuring efficiency and accuracy in data processing. A program algorithm to sum a number from one to n which is an important implementation of an algoritm in computational mathematics. This report presents the design and implementation of the program, and provides code in a specific programming language using C++ as well as using flowchart streams. The success of this program depends on the efficiency of the algorithm used. This research demonstrates the relevance and practical application of algorithms for summing numbers in software development and data processing.*

***Keywords:*** *Programming, algorithms, software.*

**ABSTRAK**

Pemrograman adalah proses membuat software komputer dengan menerapkan algoritma dan struktur data tertentu menggunakan bahasa pemrograman. Algoritma adalah metode dan tahapan sistematis yang digunakan untuk memecahkan suatu permasalahan. Struktur data adalah tempat tatanan penyimpanan data yang dibutuhkan program pada komputer. Pemahaman mendalam tentang algoritma menjadi krusial dalam pengembangan perangkat lunak dan komputasi modern, memastikan efisiensi dan ketepatan dalam pemrosesan data. Algoritma program untuk menjumlahkan bilangan satu sampai n yang merupakan implementasi algoritma yang penting dalam matematika komputasional. Laporan ini menyajikan rancangan dan implementasi program tersebut, dan memberikan kode dalam Bahasa pemograman terntentu yaitu menggunakan Bahasa C++ serta dengan memakai alur flowchart. Keberhasilan program ini bergantung pada efisiensi algoritma yang digunakan. Penelitian ini mendemostrasikan relevansi dan aplikasi praktis dari algoritma penjumlahan bilangan dalam pengembangan perangkat lunak dan pemprosesan data.

**Kata Kunci:** Pemograman, Algoritma, Perangkat Lunak.

**DAFTAR GAMBAR**

**GAMBAR 1. 8**

**GAMBAR 2. 9**

**GAMBAR 3. 10**

**GAMBAR 4. 10**

**GAMBAR 5. 11**

**GAMBAR 6. 12**

**GAMBAR 7. 12**

**GAMBAR 8. 13**

**GAMBAR 9. 14**

**DAFTAR ISI**

**KATA PENGANTAR 2**

**ABSTRAK 3**

**DAFTAR GAMBAR 4**

**DAFTAR ISI 5**

**BAB I PENDAHULUAN 6**

* 1. **LATAR BELAKANG 6**
  2. **BATASAN MASALAH 6**

**BAB II DASAR TEORI 7**

**BAB III ALGORITMA DAN FLOWCHART 13**

**BAB IV IMPLEMENTASI 15**

**BAB V PENUTUP 16**

**DAFTAR PUSTAKA 17**

**BAB I**

**PENDAHULUAN**

* 1. **LATAR BELAKANG**

Dalam konteks perkembangan teknologi informasi, pengembangan perangkat lunak menjadi aspek yang sangat penting dalam memudahkan berbagai aktivitas. Salah satu tugas umum yang seringkali diperlukan dalam pemrograman adalah menghitung jumlah dari serangkaian bilangan, terutama bilangan bulat. Dalam rangka memahami konsep dasar pengembangan perangkat lunak dan meningkatkan pemahaman dasar pemrograman, proyek ini bertujuan untuk membuat program sederhana yang dapat menjumlahkan bilangan dari 1 sampai n.

Program ini akan membantu pengguna untuk dengan cepat dan efisien menjumlahkan deret bilangan bulat tanpa harus melibatkan perhitungan manual yang memakan waktu. Selain itu, proyek ini juga memiliki tujuan untuk memberikan pengalaman praktis dalam menggunakan dasar-dasar struktur pemrograman seperti perulangan *(loop)* dan variabel.

Beberapa keuntungan dari proyek ini diantaranya, Pengenalan konsep pemrograman dasar, peningkatan keterampilan pemrograman, pemahaman algoritma penjumlahan, serta penerapan konsep pengulangan atau *(looping).* Project ini direncanakan untuk dilaksanakan dengan menggunakan bahasa pemrograman [sebutkan bahasa pemrograman yang akan digunakan], dengan fokus pada kesederhanaan dan kemudahan pemahaman. Melalui implementasi proyek ini, diharapkan pengguna dapat meningkatkan pemahaman mereka tentang dasar-dasar pemrograman dan merasakan kepuasan dalam menciptakan solusi perangkat lunak yang bermanfaat secara praktis.

* 1. **BATASAN MASALAH**

1. Program ini tidak memproses penjumlahan dari setiap bilangan.
2. Program ini menggunakan tipe data ‘int’ untuk menyimpan bilangan.
3. Program ini mungkin kurang efisien untuk bilangan sangat besar karena menggunakan pendekatan brute force.
4. Program mengasumsikan inisialisasi variabel.
5. Progam ini membutuhkan pemahaman dasar terhadap Bahasa C++ dan algoritma dasar.

**BAB II**

**DASAR TEORI**

Bahasa C++ diciptakan oleh *Bjarne Stroustrup di AT&T Bell Laboratories* awal tahun 1980-an berdasarkan C *ANSI (American National Standard Institute).* Pertama kali prototype C++ muncul sebagai C yang dipercanggih dengan fasilitas kelas. Bahasa tersebut disebut C dengan kelas (C with class). Selama tahun 1983-1984, bahasa C dengan kelas disempurnakan dengan menambahkan fasilitas pembebanlebihan operator dan fungsi yang kemudian melahirkan apa yang disebut C++. Simbol ++ merupakan operator C untuk operasi penaikan, simbol itu muncul untuk menunjukkan bahwa bahasa baru ini merupakan versi yang lebih canggih dari C. Alasan menggunakan Visual C++:

1. C++ adalah bahasa pemrograman yang populer saat ini.
2. C++ adalah bahasa pemrograman yang memiliki portabilitas tinggi.
3. C++ adalah bahasa pemrograman dengan kata kunci (keyword) sedikit.
4. C++ adalah bahasa pemrograman yang fleksibel.
5. C++ adalah bahasa pemrograman yang bersifat moduler.
6. **Algoritma**

Algoritma berarti solusi. Algoritma adalah langkah – langkah logis tertentu untuk menyelesaikan suatu masalah. Guna algoritma adalah untuk membantu seseorang dalam menyelesaikan suatu masalah berdasarkan pada pola pikirnya masing-masing. Ketika orang berbicara mengenai algoritma di bidang pemrograman, maka yang dimaksud adalah solusi dari suatu masalah yang harus dipecahkan dengan menggunakan komputer. Algoritma harus dibuat secara runut agar komputer mengerti dan mampu mengeksekusinya. Analisis kasus sangat dibutuhkan dalam membuat sebuah algoritma, misalnya proses apa saja yang sekiranya dibutuhkan untuk menyelesaikan masalah yang harus diselesaikan. Algoritma berasal dari kata algoris dan ritmis yang pertama kali diungkapkan oleh Abu Ja’far Mohammad Ibn Musa Al Khowarizmi (825M) dalam buku Al-Jabr Wa-al Muqobla. Dalam pemrograman algortima berarti suatu metode khusus yang tepat dan terdiri dari serangkaian langkah-langkah yang terstruktur dan dituliskan secara ALGORITMA & PEMROGRAMAN – Edy Budiman, M.T 9 sistematis yang akan dikerjakan untuk menyelesaikan masalah dengan bantuan komputer.

1. **Ciri – ciri algoritma:**
2. Ada masukan *(input).*
3. Ada proses.
4. Ada keluaran *(output).*
5. Memiliki perintah yang jelas dan tidak membingungkan *(ambigu).*
6. Harus mempunyai tanda akhir *(stopping role).*

1. **Sifat algoritma**:
2. Tidak menggunakan simbol atau sintaks dari suatu bahasa pemrograman.
3. Tidak tergantung pada suatu bahasa pemrograman.
4. Notasi-notasinya dapat digunakan untuk seluruh bahasa manapun.

Algoritma dapat digunakan untuk merepresentasikan suatu urutan kejadian secara logis dan dapat diterapkan di semua kejadian sehari-hari. Jadi algoritma adalah jembatan untuk mempermudah pemahaman alur kerja suatu proses.

1. **Praktik Algoritma**

Contoh kasus dalam kehidupan sehari-hari:

1. Algoritma memasak mie instan.

* Rebus air hingga mendidih.
* Masukkan mie instan ke dalam air mendidih tersebut.
* Tunggu beberapa waktu hingga mie terlihat matang.
* Jika mie sudah dirasa matang, angkat dan tiriskan.
* Campurkan bumbu-bumbu, dan aduk hingga rata.
* Mie siap disajikan/dimakan.

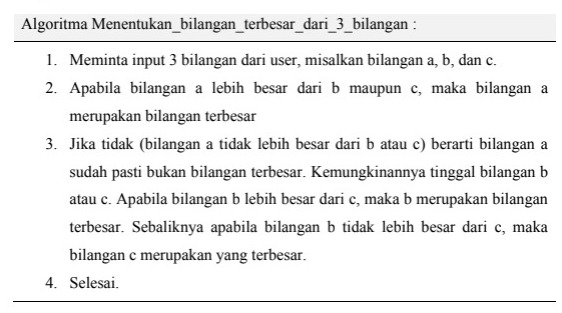
1. **Penyajian Algoritma**

Algoritma adalah independen terhadap bahasa pemrograman tertentu, artinya algoritma yang telah dibuat tidak boleh hanya dapat diterapkan pada bahasa pemrograman tertentu. Penulisan algoritma tidak terikat pada suatu aturan tertentu, tetapi harus jelas maksudnya untuk tiap langkah algoritmanya. namun pada dasarnya algoritma dibagi menjadi beberapa macam berdasarkan format penulisannya, yaitu:

1. **Deskriptif**

Algoritma bertipe deskriptif maksudnya adalah algoritma yang ditulis dalam bahasa manusia sehari-hari (misalnya Bahasa Indonesia atau Bahasa Inggris) dan dalam bentuk kalimat. Setiap langkah algoritmanya diterangkan dalam satu atau beberapa kalimat.

Contoh Algoritma Deskriptif:

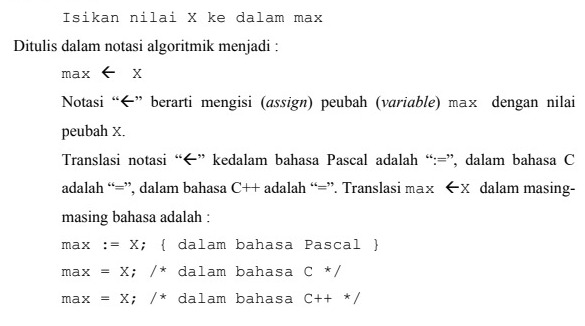


**Gambar 1. Algoritma Deskriptif**

1. **Pseudocode**

Pseudo berarti imitasi dan code berarti kode yang dihubungkan dengan instruksi yang ditulis dalam bahasa komputer (kode bahasa pemrograman). Apabila diterjemahkan secara bebas, maka pseudocode berarti tiruan atau imitasi dari kode bahasa pemrograman. Pada dasarnya, pseudocode merupakan suatu bahasa yang memungkinkan programmer untuk berpikir terhadap permasalahan yang harus dipecahkan tanpa harus memikirkan syntax dari bahasa pemrograman yang tertentu. Tidak ada aturan penulisan syntax di dalam pseudocode. Jadi pseudocode digunakan untuk menggambarkan logika urut-urutan dari program tanpa memandang bagaimana bahasa pemrogramannya. Walaupun pseudocode tidak ada aturan penulisan syntax, di dalam buku ini akan diberikan suatu aturan-aturan penulisan syntax yang cukup sederhana agar pembaca dapat lebih mudah dalam mempelajari algoritma-algoritma.

Contoh Algoritma Psedecode:



**Gambar 2. Algoritma Pseudecode**

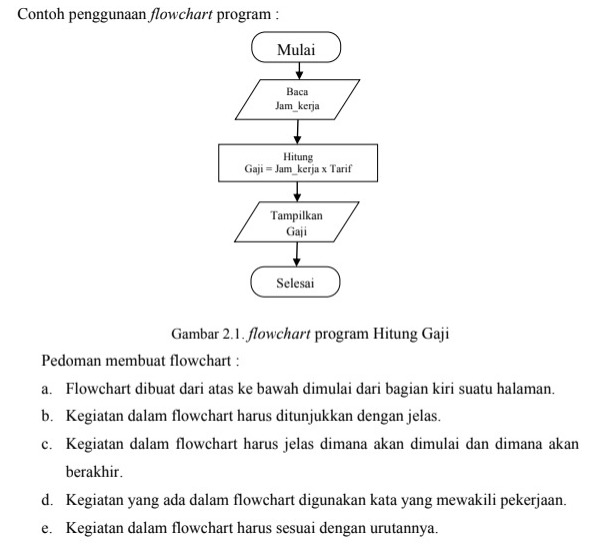
1. **Flowchart**

Dalam *Structure English* / struktur Indonesia digambarkan tahap-tahap penyelesaian masalah dengan menggunakan kata-kata (teks). Kelemahan cara ini adalah dalam penyusunan algoritma sangat dipengaruhi oleh tata bahasa pembuatnya, sehingga kadang-kadang orang lain sulit memahaminya. Oleh sebab itu kemudian dikembangkan metode yang menggambarkan tahap tahap pemecahan masalah dengan merepresentasikan symbol-simbol tertentu yang mudah dimengerti, mudah digunakan dan standar.

Contoh Algoritma Flowchart Diagram:



**Gambar 3. Algortima Flowchart Diagram**

****

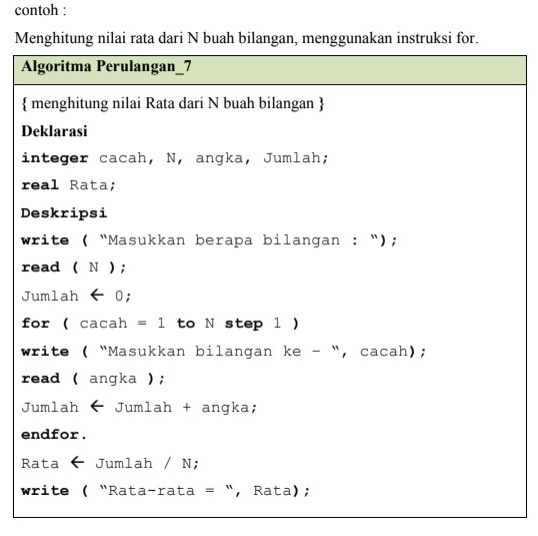
**Gambar 4. Penggugnaan Algoritma Flowchart Diagram**

1. **Intruksi Pengulangan *(Repetition)* Algoritma**

Instruksi pengulangan adalah instruksi yang dapat mengulangi pelaksanaan sederetan instruksi-instruksi lainnya berulang-kali sesuai dengan persyaratan yang ditetapkan.

Struktur instruksi pengulangan pada dasarnya terdiri atas:

1. Kondisi perulangan = suatu kondisi yang harus dipenuhi agar perulangan dapat terjadi.
2. Badan *(body)* perulangan = deretan instruksi yang akan diulang-ulang pelaksanaan-nya.
3. Pencacah *(counter)* perulangan = suatu variable yang nilainya harus berubah agar perulangan dapat terjadi dan pada akhirnya membatasi jumlah perulangan yang dapat dilaksanakan.

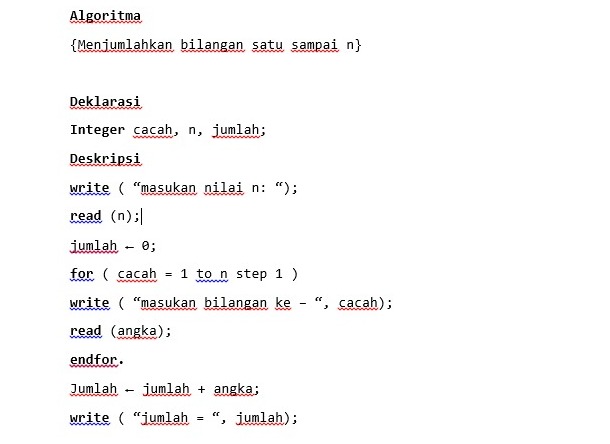


**Gambar 5. Intruksi Pengulangan For-Endfor**

**BAB III**

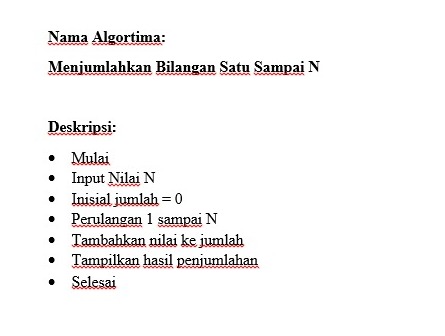
**ALGORITMA DAN FLOWCHART**

1. **Algoritma**

****

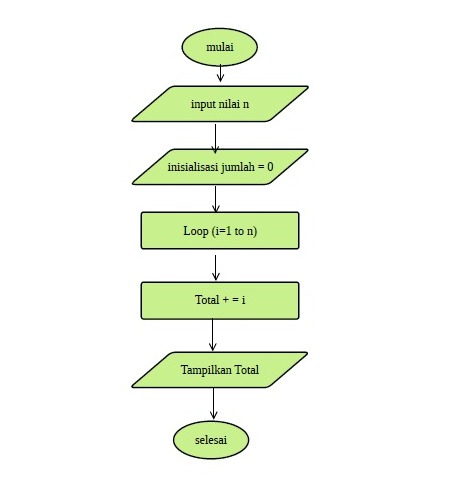
**Gambar 6. Algoritma Penjumlahan Bilangan 1 sampai N**

1. **Pseudecode**

****

**Gambar 7. Pseudecode Penjumlahan Bilangan 1 Sampai N**

1. **Flowchart**

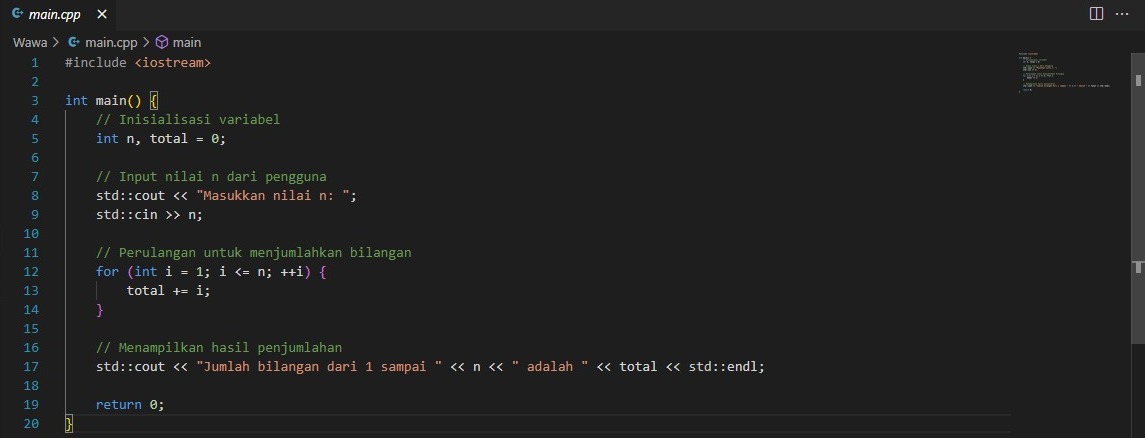
****

**Gambar 8. Flowchart Penjumlahan Bilangan 1 sampai N**

**BAB IV**

**IMPLEMENTASI**

1. **Kode Program**

****

**Gambar 9. Kode Program Penjumlahan Bilangan 1 sampai N**

**BAB V**

**PENUTUP**

Sebagai penutup laporan ini menceminkan pentingnya pengembangan program pengecekan bilangan prima. Dengean adanya aplikasi VSCODE, diharapkan dapat memberikan kontribusi positif dalam memahami dan mengidentifikasi bilangan prima secara efisien. Dengan demikian, kita dapat meningkatkan efektivitas dalam pemrosesan data matematika dan mengoptimalkan penggunaan sumber daya komputasi.

Terima kasih atas perhatian dan harapan program ini dapat bermanfaat dalam konteks penelitian dan aplikasi VSCODE yang menggunakan dan mengembangkan oleh Bahasa Program C++.

**DAFTAR PUSTAKA**

#### Buku Algoritma dan Pemograman.pdf

Buku penggunaan aplikasi Visual C++ untuk dasar pemrograman komputer

#### <https://eprints.utdi.ac.id/5635/1modul%20Algoritma%20TI%20Pertemuan%201-14.pdf>

#### <https://www.google.com/search?q=gambar%2Bsimbol-simbol%2Bflowchart%2Bdiagram&oq>

Jurnal Laporan Praktikum Dasar Pemograman Komputer.docx.UM2014