**LAPORAN HASIL PRAKTIKUM**

**ALGORITMA DAN PEMOGRAMAN I**

****

**Nama : Jordan Setiawan Nanyan**

**NIM : 213020503058**

**Kelas : D**

**Modul :** **PERCABANGAN (DECISION)**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PALANGKA RAYA**

**2021**

# BAB 1

**TUJUAN DAN LANDASAN TEORI**

* 1. **Tujuan :** 
     1. Dapat mengetahui perbedaan penggunaan if-then-else dan case..of dalam setiap permasalahan.
     2. Dapat menggunakan if-then-else dan case of dalam menyelesaikan sebuah permasalahan yang dituangkan dalam sebuah program.
  2. **Landasan Teori :**

Penyeleksi kondisi atau pernyataan kondisi (conditional statement) merupakan suatu pernyataan yang menganalisa suatu keadaan dan mengambil keputusan berdasarkan pada hasil analisa itu. Hasil dari penyeleksian jika bernilai benar maka akan dikerjakan instruksi tertentu. Jika salah maka akan dikerjakan instruksi yang lain.

Pada struktur percabangan, program akan berpindah urutan pelaksanaan jika suatu kondisi yang disyaratkan terpenuhi. Pada Flowchart, simbol flowchart Decision yang digunakan pada proses ini. Simbol decision akan berisi pernyataan yang akan diuji kebenarannya. Nilai hasil pengujian akan menentukan cabang mana yang akan ditempuh.

Bentuk strukturnya adalah:

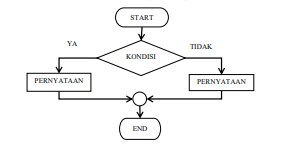
If *Kondisi* then

*Statement 1*

Else

*Statement 2*

Dalam bentuk Flowchart akan menjadi sebagai berikut:



Fungsi/kegunaan dari struktur pemilihan adalah untuk membuat struktur menu, memvalidasi inputan, dan mencegah error.

Konsep dasar dan bentuk struktur dari percabangan dalam bahasa pemograman Pascal adalah sebagai berikut:

1. If – Then

If *Kondisi* Then begin

*Kode Program*

end;

1. If – Then – Else

|  |
| --- |
| If *Kondisi* Then begin  *Kode Program 1*  end Else begin  *Kode Program 2*  End; |

1. Case – Of
   1. Bentuk Case - Of

|  |  |
| --- | --- |
| Case *kondisi* of nilai1 | : |
| *statemen1*; nilai2 | : |
| *statemen2*; nilai3 | : |
| *statemen3*; nilai4  *statemen4;*  *..................*  End; | : |

* 1. Bentuk Case-Of-Else

|  |  |
| --- | --- |
| Case *kondisi* of nilai1  *statemen1*;  …………………  nilaiN :  statemenN;  Else  Statemen Lain;  End; | : |

* 1. **Tugas**

1. Buatlah program untuk menentukan bilangan bulat terbesar dan yang terkecil dari tiga buah bilangan bulat!
2. Buatlah program untuk membaca nomor bulan (integer), lalu menuliskan nama bulan sesuai dengan angka bulannya. Misalnya jika dibaca angka 9, maka tercetak agustus.
3. Simulasikan sebuah kalkulator sederhana untuk melakukan operasi aritmatika sebagai berikut : dibaca operand 1, operator, dan operand 2.

Tentukan hasil operasi aritmatika :

Operand 1 Operator Operand 2

7 “+” 3

Maka hasilnya adalah 10

1. Buatlah program untuk menentukan Harga Mutu serta keterangan Lulus/Tidak Lulus dari nilai akhir sebuah mata kuliah. Dengan ketentuan sebagai berikut.

|  |  |  |
| --- | --- | --- |
| **Nilai** | **Harga Mutu** | **Keterangan** |
| **> 80** | **A** | **Lulus** |
| **75-79** | **B+** | **Lulus** |
| **70-74** | **B** | **Lulus** |
| **65-69** | **C+** | **Lulus** |
| **60-64** | **C** | **Lulus** |
| **50-59** | **D** | **Tidak Lulus** |
| **< 50** | **E** | **Tidak Lulus** |

Misalkan diberi masukan nilai 73 maka akan muncul output:

Harga mutu = B

Keterangan = Lulus

1. Buatlah program kalkulator sederhana menggunakan percabangan case..of seperti dibawah ini

|  |
| --- |
| ========= My Calculator  ========   1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian pilih menu : 1 masukkan bilangan 1 = 3 masukkan bilangan 2 = 4 hasil perhitungan = 3 + 4 = 7   Terima Kasih |

**BAB 2**

**LANGKAH PENYELESAIAN DAN PEMBAHASAN TUGAS PRAKTIKUM**

* 1. **Langkah Penyelesaian dan Penjelasan**
     1. Langkah Penyelesaian dan Penjelasan Tugas No 1

Tugas No.1 meminta untuk membuat sebuah program yang dapat menentukan bilangan bulat terbesar dari tiga bilangan bulat yang diinputkan. Algoritma programnya akan ditulis terlebih dahulu, dan akan terlihat seperti ini :

program Perbandingan\_3\_angka;

uses crt;

var

  a, b, c: real;

begin

  clrscr;

  writeln('Program Perbandingan 3 variabel');

  writeln('-----------------------------------');

  write('masukkan bilangan pertama: '); readln(a);

  write('masukkan bilangan kedua  : '); readln(b);

  write('masukan  bilangan ketiga : '); readln(c);

  if (a > b) and  (a > c) then

  begin

    writeln('bilangan pertama adalah bilangan terbesar')

  end

  else if (b > a) and (b > c) then

  begin

    writeln('bilangan kedua adalah bilangan terbesar')

  end

  else

  begin

    writeln('bilangan ketiga adalah bilangan terbesar');

  end;

end.

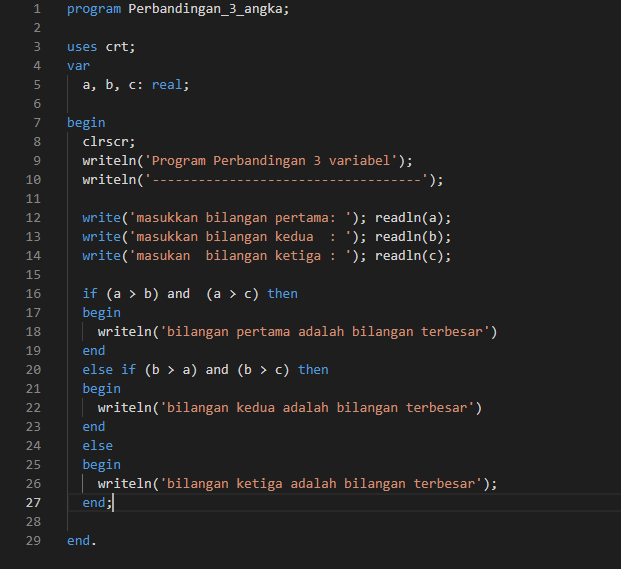
# BAB 3

## KESIMPULAN

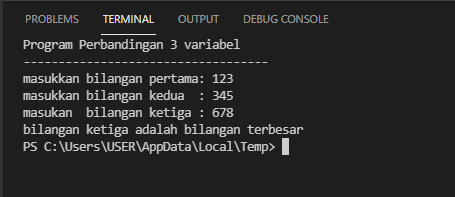
.

Algoritma tersebut memiliki tiga variabel yang merepresentasikan tiga bilangan bulat yang akan dibandingkan, dan ketiga variabel tersebut merupakan variabel berjenis real karena variabel real memiliki range yang lebih luas daripada variabel integer. Dan karena programnya hanya membandingkan bilangan bulat menggunakan jenis algoritma if then else, maka jenis variabel yang digunakan tidak terlalu berpengaruh.

Jika algoritma tersebut dimasukan kedalam aplikasi Visual Studio Code maka akan terlihat seperti berikut:

Gambar Program Perbandingan\_3\_angka Di Visual Studio Code

Dari program tersebut maka output dari Visual Studio Code akan terlihat seperti ini seperti berikut:



Gambar Output Dari Program Perbandingan\_3\_angka di Visual Studio Code.

Dari program diatas dapat disimpulkan beberapa hal, penggunaan struktur if then else cukup mudah digunakan dan sangat berguna untuk program yang memiliki percabangan dengan kondisi yang rumit atau kompleks. Karena itu struktur if then else sangat mudah digunakan, yang perlu dilakukan hanyalah memasukan kondisi yang ingin dipenuhi dan menuliskan sebuah kode program untuk jika kondisi dari struktur if then terpenuhi. Karena program di atas adalah sebuah program perbandingan, maka yang diperlukan hanyalah kondisi perbandingan di antara bilangan bulat yang diinputkan. Tidak perlu melakukan proses lain.

* + 1. Langkah Penyelesaian dan Penjelasan Tugas No 2

Tugas No.2 meminta untuk membuat sebuah program yang dapat menuliskan nama bulan pada outputnya ketika nomor dari bulan tersebut diinputkan. Algoritma programnya akan ditulis terlebih dahulu, dan akan terlihat seperti ini :

program menentukan\_nama\_bulan;

uses crt;

var a: integer;

begin

clrscr;

write('Masukan Nomor Bulan: '); readln(a);

write('Merupakan ');

case (a) of

1: writeln('Bulan Januari');

2: writeln('Bulan Febuari');

3: writeln('Bulan Maret');

4: writeln('Bulan April');

5: writeln('Bulan Mei');

6: writeln('Bulan Juni');

7: writeln('Bulan Juli');

8: writeln('Bulan Agustus');

9: writeln('Bulan September');

10: writeln('Bulan Oktober');

11: writeln('Bulan November');

12: writeln('Bulan Desember');

else writeln('Nomor Bulan Tidak Valid');

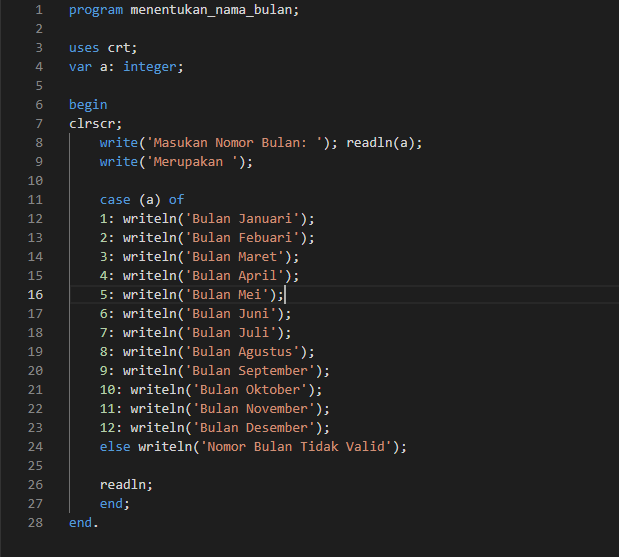
readln;

end;

end.

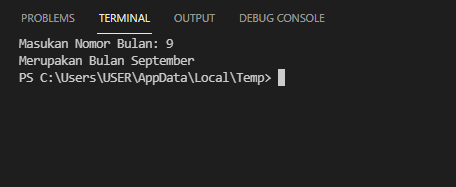
Di algoritma di atas programnya menggunakan satu variabel yang berjenis variabel integer dan struktur percabangan case of. Hal tersebut dikarenakan oleh beberapa alasan, karena struktur case of tidak menggunakan tanda kutip untuk angka maka integer dapat digunakan, jika menggunakan variabel berjenis lain maka output akan memberikan error yang bertuliskan ‘Constant and CASE types do not match’. Jika nomor bulan menggunaka tanda kutip maka kita harus menggunakan variabel string, agar error tersebut tidak muncul. Di akhir dari struktur case of juga ada kode bertuliskan ‘else writeln('Nomor Bulan Tidak Valid');’, jika angka yang diinputkan tidak sesuai dengan nomor bulan. Untuk penggunaan struktur case of, dapat dilihat bahwa lebih mudah dari Struktur If Then Else, karena lebih singkat dan mudah dibaca.

Jika dituliskan di Visual Studio Code akan terlihat seperti ini:



Gambar Program menentukan\_nama\_bulan di Visual Studio Code.

Dan output dari program tersebut jika dijalankan adalah:



Gambar Output Untuk Program menentukan\_nama\_bulan di Visual Studio Code.

Sebenarnya penulisan program di atas dapat menggunakan struktur case of atau if then else, semua tergantung dengan kegunaan dan situasi yang diperlukan. Struktur case of sangat menguntungkan jika percabangan di dalam program termasuk cukup simpel tetapi agak panjang, karena hal tersebut penggunaan if then else akan memakan banyak penulisan kode dan mencadi terlalu rumit dari case of.

* + 1. Langkah Penyelesaian dan Penjelasan Tugas No 3

Tugas No. 3 meminta untuk mensimulasikan sebuah kalkulator sederhada untuk operasi aritmatika dimana input dari program adalah operand 1, operator, dan operand 2. Algoritma tersebut jika kita tuliskan akan terlihat seperti ini:

program simulasi\_kalkulator;

uses crt;

var a, b, c:real;

    aritmatika :char;

begin

    clrscr;

    writeln('Masukkan angka pertama');

    writeln('Masukkan operator (+, -, /, \*)');

    writeln('Masukkan Angka kedua');

    writeln('------------------------------------------------');

    write('Angka Pertama      : '); readln(a);

    write('Masukan Operatornya: '); readln(aritmatika);

    write('Angka Kedua        : '); readln(b);

    case (aritmatika) of

        '+': c:= a+b;

        '-': c:= a-b;

        '\*': c:= a\*b;

        '/': c:= a/b;

        else writeln('Operator Tidak Valid');

    end;

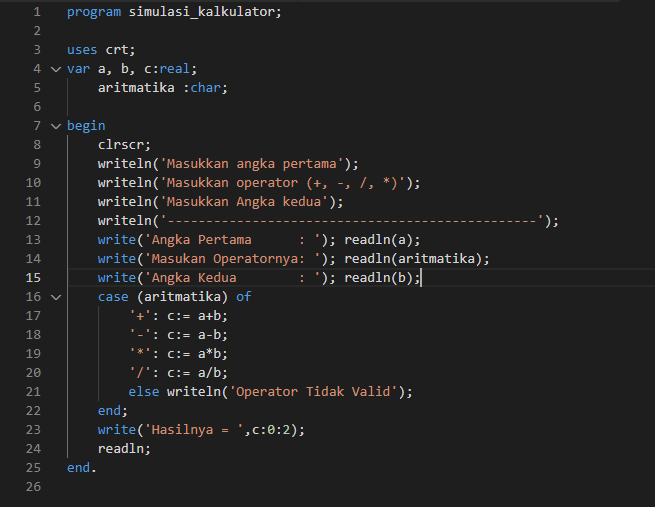
    write('Hasilnya = ',c:0:2);

    readln;

end.

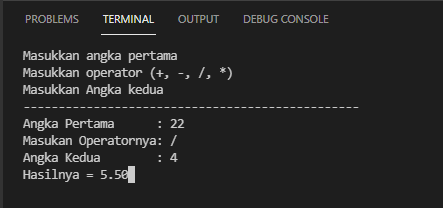
Algoritma di atas menggunakan struktur percabangan case of, yang bekerja sangat baik untuk permasalahan simpel dan mudah. Algoritma di atas menggunakan 4 variabel yaitu (a, b, c, dan aritmatika) yang dimana tiga dari empat variabel tersebut berjenis real, dan variabel ‘aritmatika’ berjenis char. Variabel yang berjenis real berguna untuk angka yang akan di inputkan dan hasil dari perhitungan aritmatikanya, sedangkan variabel ‘aritmatika’ dipakai untuk menentukan percabangan case of, proses yang mana yang akan dilaksanakan dengan memasukan lambang perhitungan aritmatika yang ingin dilaksanakan.

Jika dituliskan di Visual Studio Code akan terlihat seperti ini:



Gambar Program simulasi\_kalkulator di Visual Studio Code

Dan output dari program tersebut jika dijalankan adalah:



Gambar Output Untuk Program Simulasi\_kalkulator di Visual Studio Code.

Sebenarnya penulisan program di atas dapat menggunakan struktur case of atau if then else, tetapi karena jenis program yang ingin dibuat cukup mudah, maka penggunaan struktur case of lebih mudah daripada if then else. Penulisan program akan lebih singkat danlebih mudah dipahami.

* + 1. Langkah Penyelesaian dan Penjelasan Tugas No 4

Di tugas No. 4 diberikan tugas untuk membuat sebuah program yang menentukan harga mutu dan menuliskan keterangannya di output. Algoritma tersebut jika kita tuliskan akan terlihat seperti ini:

program menentukan\_harga\_mutu;

uses crt;

var a: integer;

begin

clrscr;

writeln('Program Menentukan Harga Mutu Nilai');

writeln('--------------------------------------');

write('Masukan Nilai: '); readln(a);

if (a>=80) then

begin

writeln('Harga Mutu = A');

writeln('Keterangan = Lulus');

end

else if (a>=75) and (a<=79) then

begin

writeln('Harga Mutu = B+');

writeln('Keterangan = Lulus');

end

else if (a>=70) and (a<=74) then

begin

writeln('Harga Mutu = B');

writeln('Keterangan = Lulus');

end

else if (a>=65) and (a<=69) then

begin

writeln('Harga Mutu = C+');

writeln('Keterangan = Lulus');

end

else if (a>=60) and (a<=64) then

begin

writeln('Harga Mutu = C');

writeln('Keterangan = Lulus');

end

else if (a>=50) and (a<=59) then

begin

writeln('Harga Mutu = D');

writeln('Keterangan = Tidak Lulus');

end

else if (a<50) then

begin

writeln('Harga Mutu = D');

writeln('Keterangan = Tidak Lulus');

end

else

begin

writeln('Nilai Tidak Valid');

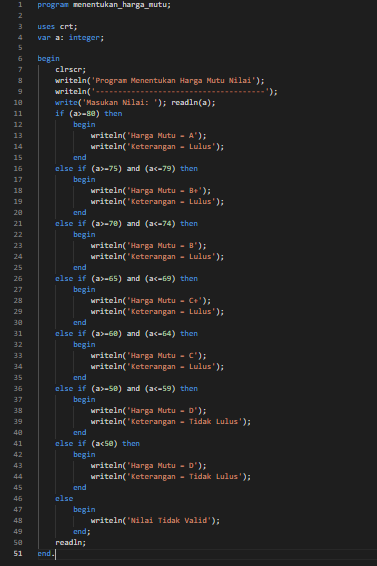
end;

readln;

end.

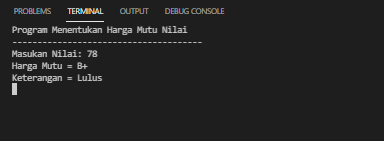
Agoritma di atas menggunakan struktur percabangan if then else untuk menentukan harga mutu nilainya. Hal ini disebabkan karena struktur if then else lebih bagus dibandingkan dengan case of jika program yang dibuat memiliki beberapa intruksi ketika kondisi dari if then else terpenuhi.

Jika dituliskan di dalam Visual Studio Code maka akan terlihat seperti ini:



Gambar Program menentukan\_harga\_mutu di Visual Studio Code.

Dan output dari program tersebut akan terlihat seperti ini:



Gambar Output Program konversidetik di Visual Studio Code

Dari percobaan di atas, dapat dipelajari bahwa penggunaan struktur percabangan if then else adalah struktur yang cukup panjang dan lebih rumit dari case of. Tetapi struktur if then else lebih mudah digunakan untuk program yang memberikan intruksi lebih dari satu ketika kondisi yang ditetapkan terpenuhi.

* + 1. Langkah Penyelesaian dan Penjelasan Tugas No 5

Tugas No. 5 memberi tugas untuk membuat program kalkulator sederhana menggunakan percabangan case..of seperti dibawah ini

|  |
| --- |
| ========= My Calculator ========   1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian   pilih menu : 1 masukkan bilangan 1 = 3 masukkan bilangan 2 = 4 hasil perhitungan = 3 + 4 = 7  Terima Kasih |

Algoritma yang ditulis akan terlihat seperti ini:

program My\_calculator;

uses crt;

var a, b, c:real;

d:char;

//Mendefinisikan Prosedur

procedure pertambahan;

begin

writeln;

writeln('Kalkulator Pertambahan');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a+b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

procedure pengurangan;

begin

writeln;

writeln('Kalkulator Pengurangan');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a-b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

procedure perkalian;

begin

writeln;

writeln('Kalkulator Perkalian');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a\*b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

procedure pembagian;

begin

writeln;

writeln('Kalkulator Pembagian');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a/b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

begin

clrscr;

// Memanggil Prosedur

writeln('Menu My calculator');

Writeln('-----------------------');

writeln('1. Pertambahan');

writeln('2. Pengurangan');

writeln('3. Perkalian');

writeln('4. Pembagian');

write('Pilih Salah Satu Operasi Aritmatikannya: '); readln(d);

case (d) of

'1': pertambahan;

'2': pengurangan;

'3': perkalian;

'4': pembagian;

end;

readln;

end.

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

procedure pengurangan;

begin

writeln;

writeln('Kalkulator Pengurangan');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a-b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

procedure perkalian;

begin

writeln;

writeln('Kalkulator Perkalian');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a\*b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

procedure pembagian;

begin

writeln;

writeln('Kalkulator Pembagian');

write('Masukan Bilangan Pertama: '); readln(a);

write('Masukan Bilangan Kedua: '); readln(b);

c:= a/b;

writeln('Hasil Operasinya adalah: ',c:0:2);

end;

begin

clrscr;

// Memanggil Prosedur

writeln('Menu My calculator');

Writeln('-----------------------');

writeln('1. Pertambahan');

writeln('2. Pengurangan');

writeln('3. Perkalian');

writeln('4. Pembagian');

write('Pilih Salah Satu Operasi Aritmatikannya: '); readln(d);

case (d) of

'1': pertambahan;

'2': pengurangan;

'3': perkalian;

'4': pembagian;

end;

readln;

end.

Writeln('-----------------------');

writeln('1. Pertambahan');

writeln('2. Pengurangan');

writeln('3. Perkalian');

writeln('4. Pembagian');

write('Pilih Salah Satu Operasi Aritmatikannya: '); readln(d);

case (d) of

'1': pertambahan;

'2': pengurangan;

'3': perkalian;

'4': pembagian;

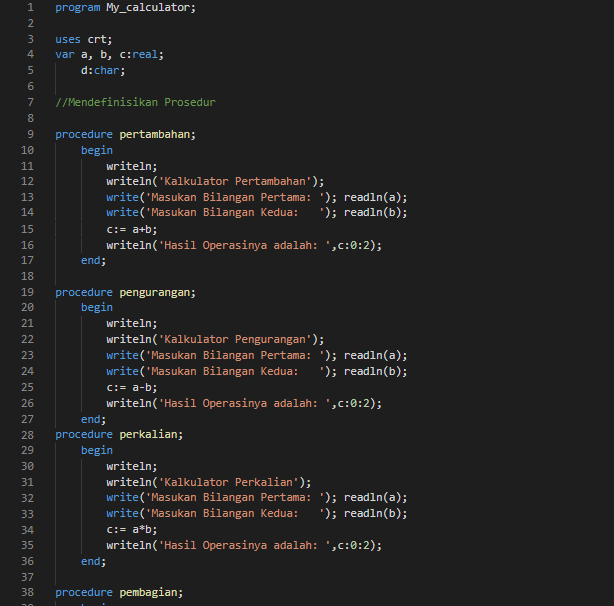
end;

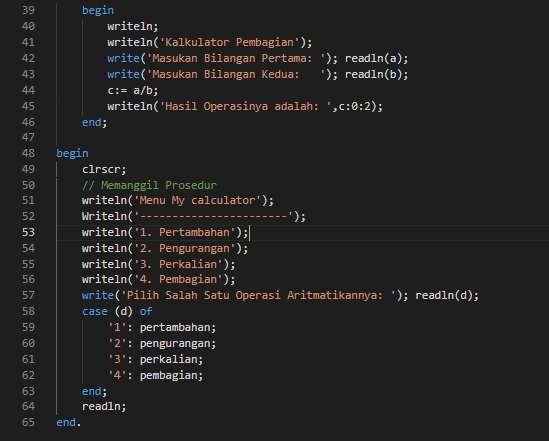
readln;

end.

Algoritma di atas menggunakan apa yang disebut dengan program procedure untuk memberikan output yang terlihat seperti menu, seperti yang diperintahkan oleh soal. Hal pertama yang perlu dilakukan adalah mendefinisikan procedurenya dan kemudian memanggil procedure tersebut dengan struktur percabangan case of, sesuai dengan operasi aritmatika yang kita inginkan.

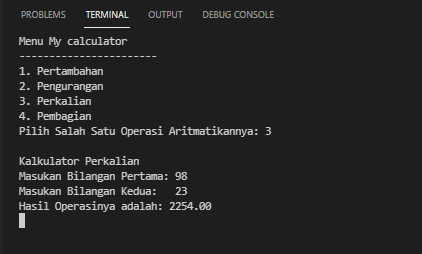
Jika dimasukan kedalam Visual Studio Code maka akan terlihat seperti ini:





Gambar Program My\_calculator di Visual Studio Code

Dan output yang akan kita dapatkan adalah:



Gambar Output Program My\_calculator di Visual Studio Code

Dari percobaan di atas ada beberapa hal yang dapat dipelajari. Struktur percabangan case of dapat memberikan output yang terlihat seperti sebuah menu dengan menggunakan sedikit kreatifitas, seperti program procedure yang dapat dipanggil ketika memilih operasi aritmatika yang diinginkan. Variabel yang digunakan sebenarnya cukup simpel, yaitu tiga variabel real untuk operasi aritmatikanya, dan satu variabel lagi yang berjenis char untuk mendeklarasikan struktur percabangan case of nya.

**KESIMPULAN**

Di modul tiga, yang diharapkan adalah mahasiswa dapat memahami dua struktur percabangan, yaitu case of dan if then else dan mengetahui perbedaan dari kedua struktur tersebut. kedua struktur tersebut dapat menghasilkan output yang sama tetapi dengan cara yang jauh berbeda, struktur case berguna untuk melakukan proses yang simpel dan mudah kecuali kita menggunakan program procedure agar memberi struktur case of lebih banyak kegunaan dan fleksibilitas, sedangkan struktur if then else memiliki banyak keunggulan dan yang terutama adalah struktur ini dapat dibuat menjadi sebesar apapun, sesuai dengan apa yang diinginkan dengan sebanyak percabangan yang diinginkan, struktur if then else juga dapat memiliki cabang di dalam cabang karena fleksibilitas dari struktur ini yang membolehkan hal tersebut.

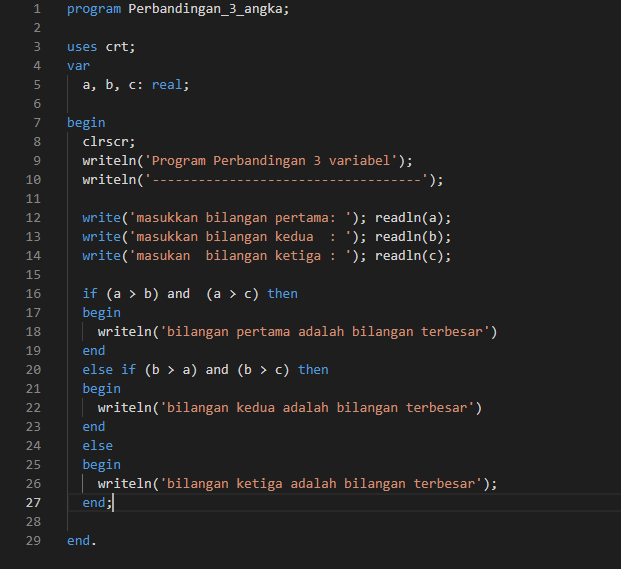
# DAFTAR PUSTAKA

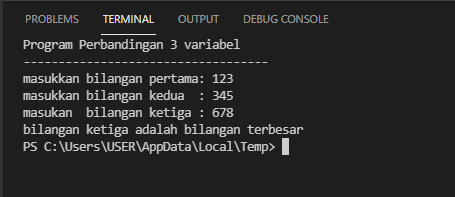
Percabangan dengan CASE OF pada Pascal https://forum.lazarus.freepascal.org/index.php?topic=41823.0 [Diakses pada tanggal 21 Oktober 2021]

Error: Constant and CASE types do not match with string

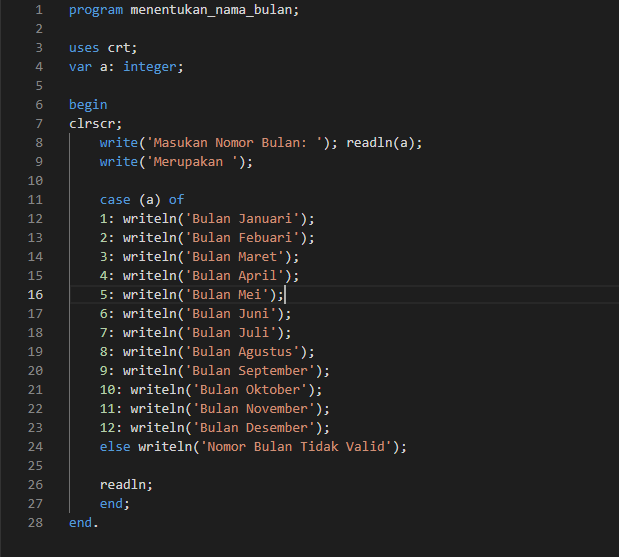
https://invasikode.com/case-of-pascal/ [Diakses pada tanggal 21 Oktober 2021]

# LAMPIRAN

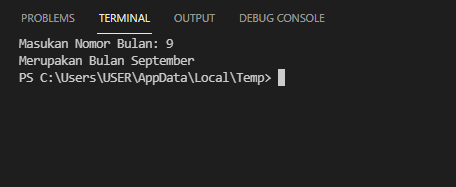
Gambar Program Perbandingan\_3\_angka Di Visual Studio Code



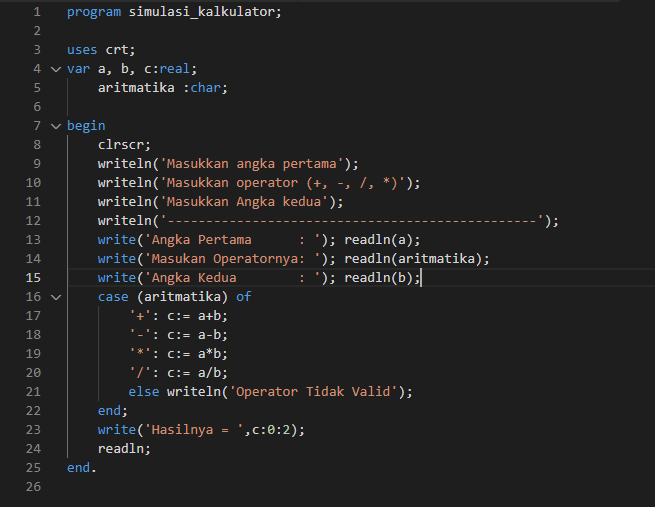
Gambar Output Dari Program perbandingan\_3\_angka di Visual Studio Code.



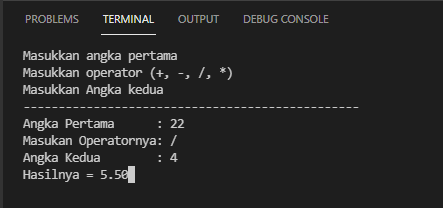
Gambar Program menentukan\_nama\_bulan di Visual Studio Code.



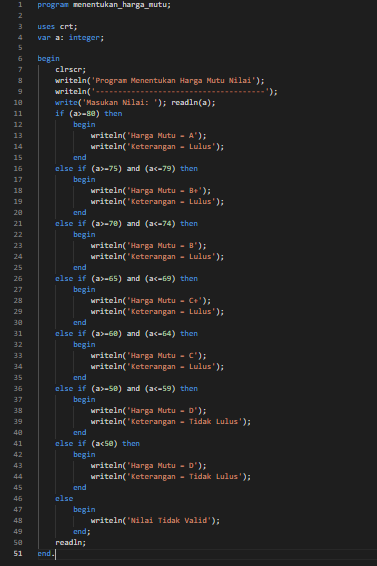
Gambar Output Untuk Program menentukan\_nama\_bulan di Visual Studio Code.



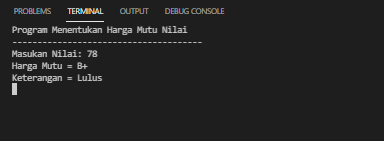
Gambar Program simulasi\_kalkulator di Visual Studio Code



Gambar Output Untuk Program Simulasi\_kalkulator di Visual Studio Code.



Gambar Program menentukan\_harga\_mutu di Visual Studio Code.



Gambar Output Program konversidetik di Visual Studio Code

**LAPORAN RENCANA PRAKTIKUM**

**ALGORITMA DAN PEMOGRAMAN I**



**Nama : Jordan Setiawan Nanyan**

**NIM : 213020503058**

**Kelas : D**

**Modul : PERCABANGAN (DECISION)**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PALANGKA RAYA**

**2021**

**MODUL III**

**PERCABANGAN (DECISION)**

**BAB 1**

**TUJUAN DAN LANDASAN TEORI**

* 1. **Tujuan :** 
     1. Dapat mengetahui perbedaan penggunaan if-then-else dan case..of dalam setiap permasalahan.
     2. Dapat menggunakan if-then-else dan case of dalam menyelesaikan sebuah permasalahan yang dituangkan dalam sebuah program.
  2. **Landasan Teori :**

Penyeleksi kondisi atau pernyataan kondisi (conditional statement) merupakan suatu pernyataan yang menganalisa suatu keadaan dan mengambil keputusan berdasarkan pada hasil analisa itu. Hasil dari penyeleksian jika bernilai benar maka akan dikerjakan instruksi tertentu. Jika salah maka akan dikerjakan instruksi yang lain.

Pada struktur percabangan, program akan berpindah urutan pelaksanaan jika suatu kondisi yang disyaratkan terpenuhi. Pada Flowchart, simbol flowchart Decision yang digunakan pada proses ini. Simbol decision akan berisi pernyataan yang akan diuji kebenarannya. Nilai hasil pengujian akan menentukan cabang mana yang akan ditempuh.

Bentuk strukturnya adalah:

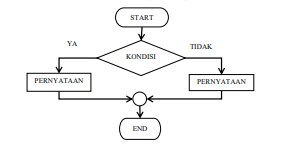
If *Kondisi* then

*Statement 1*

Else

*Statement 2*

Dalam bentuk Flowchart akan menjadi sebagai berikut:



Fungsi/kegunaan dari struktur pemilihan adalah untuk membuat struktur menu, memvalidasi inputan, dan mencegah error.

Konsep dasar dan bentuk struktur dari percabangan dalam bahasa pemograman Pascal adalah sebagai berikut:

1. If – Then

If *Kondisi* Then begin

*Kode Program*

end;

1. If – Then – Else

|  |
| --- |
| If *Kondisi* Then begin  *Kode Program 1*  end Else begin  *Kode Program 2*  End; |

1. Case – Of
   1. Bentuk Case - Of

|  |  |
| --- | --- |
| Case *kondisi* of nilai1 | : |
| *statemen1*; nilai2 | : |
| *statemen2*; nilai3 | : |
| *statemen3*; nilai4  *statemen4;*  *..................*  End; | : |

* 1. Bentuk Case-Of-Else

|  |  |
| --- | --- |
| Case *kondisi* of nilai1  *statemen1*;  …………………  nilaiN :  statemenN;  Else  Statemen Lain;  End; | : |

* 1. **Tugas**

1. Buatlah program untuk menentukan bilangan bulat terbesar dan yang terkecil dari tiga buah bilangan bulat!
2. Buatlah program untuk membaca nomor bulan (integer), lalu menuliskan nama bulan sesuai dengan angka bulannya. Misalnya jika dibaca angka 9, maka tercetak agustus.
3. Simulasikan sebuah kalkulator sederhana untuk melakukan operasi aritmatika sebagai berikut : dibaca operand 1, operator, dan operand 2.

Tentukan hasil operasi aritmatika :

Operand 1 Operator Operand 2

7 “+” 3

Maka hasilnya adalah 10

1. Buatlah program untuk menentukan Harga Mutu serta keterangan Lulus/Tidak Lulus dari nilai akhir sebuah mata kuliah. Dengan ketentuan sebagai berikut.

|  |  |  |
| --- | --- | --- |
| **Nilai** | **Harga Mutu** | **Keterangan** |
| **> 80** | **A** | **Lulus** |
| **75-79** | **B+** | **Lulus** |
| **70-74** | **B** | **Lulus** |
| **65-69** | **C+** | **Lulus** |
| **60-64** | **C** | **Lulus** |
| **50-59** | **D** | **Tidak Lulus** |
| **< 50** | **E** | **Tidak Lulus** |

Misalkan diberi masukan nilai 73 maka akan muncul output:

Harga mutu = B

Keterangan = Lulus

1. Buatlah program kalkulator sederhana menggunakan percabangan case..of seperti dibawah ini

|  |
| --- |
| ========= My Calculator  ========   1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian pilih menu : 1 masukkan bilangan 1 = 3 masukkan bilangan 2 = 4 hasil perhitungan = 3 + 4 = 7   Terima Kasih |