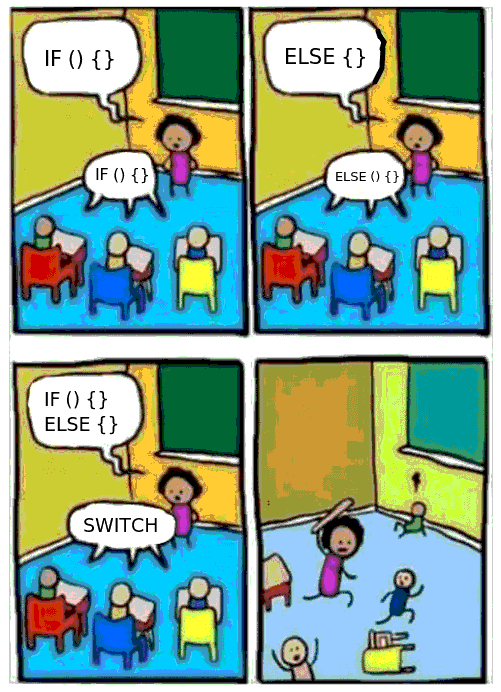
# Tujuan

## **MODUL 4** **PEMILIHAN**



* 1. Pratikan dapat memahami tentang pemilihan/percabangan didalam Bahasa Pemrograman C.
  2. Pratikan dapat mengetahui waktu yang tepat dalam penggunaan pemilihan/percabangan pada studi kasus tertentu.

# Penjelasan

Pemilihan atau percabangan merupakan sebuah algoritma kondisi yang mana terdapat persyaratan tertentu yang akan dieksekusi dan memenuhi suatu kondisi, pada umumnya saat membuat program yang lebih kompleks akan sering menjumpai suatu permasalahan yang hanya dapat diselesaikan dengan menggunakan pemilihan/percabangan. Pada pemilihan sendiri memiliki dua nilai yaitu **true** dan **false**, suatu aksi hanya akan dilakukan jika nilai dari kondisinya bernilai true atau benar. Kedua nilai ini selalu mempengaruhi bagaimana nantinya program yang dibuat akan bekerja.

# OPERATOR BOOLEAN

Operator Boolean adalah operator yang digunakan untuk membandingkan suatu kondisi/persyaratan didalam pengkodisian, hasil yang didapatkan dari operator Boolean bernilai true atau false. Bentuk penggunaan dari operator Boolean,

Contoh : Kondisi 1 <operator> Kondisi 2.

|  |  |
| --- | --- |
| Operator | Keterangan |
| ==  (Sama dengan) | Bernilai True jika ruas kiri dan kanan memiliki nilai yang sama. Contoh : a = 1, b =1  a==b –> TRUE |
| !=  (Tidak sama dengan) | Benilai True jika nilai di ruas kiri dan kanan berbeda.  Contoh: a=2, b=2 a!=b –> FALSE |
| <  (Kecil dari) | Bernilai True jika nilai di ruas kiri lebih kecil dari ruas  kanan.  Contoh: a=3, b=7  a<b –> TRUE |

|  |  |
| --- | --- |
| >  (Besar dari) | Bernilai True jika nilai di ruas kanan lebih besar dari nilai di ruas kiri.  Contoh: a=2, b=3  a>b –> FALSE |
| <=  (Kecil dari sama dengan) | Benilai True jika nilai di ruas kiri lebih kecil atau sama dengan ruas kanan.  Contoh: a=2, b=5, c=5 a<=b –> TRUE  c<=b –> TRUE |
| >=  (Besar dari sama dengan) | Bernilai True jika nilai di ruas kanan lebih besar atau sama dengan ruas kiri.  Contoh: a=2, b=5, c=5 a>=b –> FALSE  b>=c –> FALSE |
| !  (Negasi) | Bernilai True jikai nilai komplemen atau kebalikan dari suatu kondisi.  Contoh: a=2, b=1, c=5  !(b==a) –> TRUE  !(b<a) –> FALSE |
| &&  (Dan) | Benilai True jika nilai di ruas kiri dan kanan keduanya bernilai True.  Contoh: a=2, b=1, c=5 a>=b && b<c –> TRUE  a==c && c>b FALSE |
| ||  (Atau) | Bernilai True jika nilai di ruas kiri dan kanan salah satunya bernilai True.  Contoh: a=2, b=1 a>b || a==b –> TRUE  b>a || b==a –> FALSE |

# PEMILIHAN

Pada pemilihan/percabangan terdapat 3 jenis umum pemilihan yaitu IF-ELSE, SWITCH, dan Operator Ternary. Berikut merupakan contoh dari 3 pemilihan tersebut.

## IF-ELSE

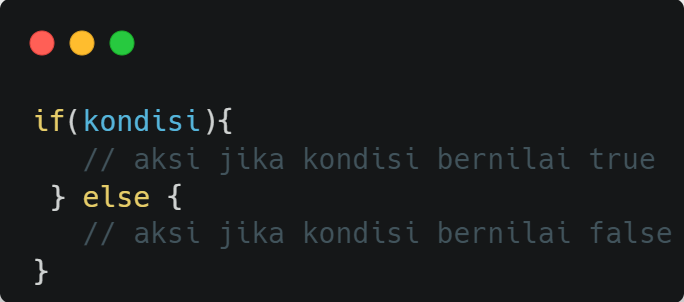
* 1. Satu kasus



**Gambar 1. Satu Kasus (If)**

Aksi akan dieksekusi jika kondisi tersebut terpenuhi, namun jika kondisi tidak terpenuhi maka aksi tidak akan tereksekusi.

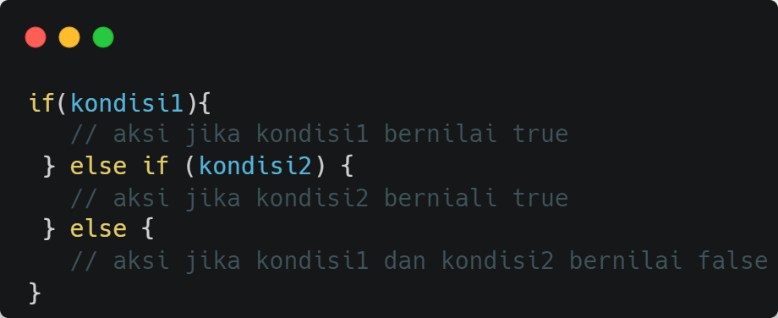
* 1. Dua Kasus



**Gambar 2. Dua Kasus (If - Else)**

Aksi pada bagian if akan dijalankan jika kondisi tersebut memenuhi syarat, dan jika kondisi tidak memenuhi syarat maka aksi pada bagian else akan dijalankan.

* 1. Tiga atau lebih banyak kasus

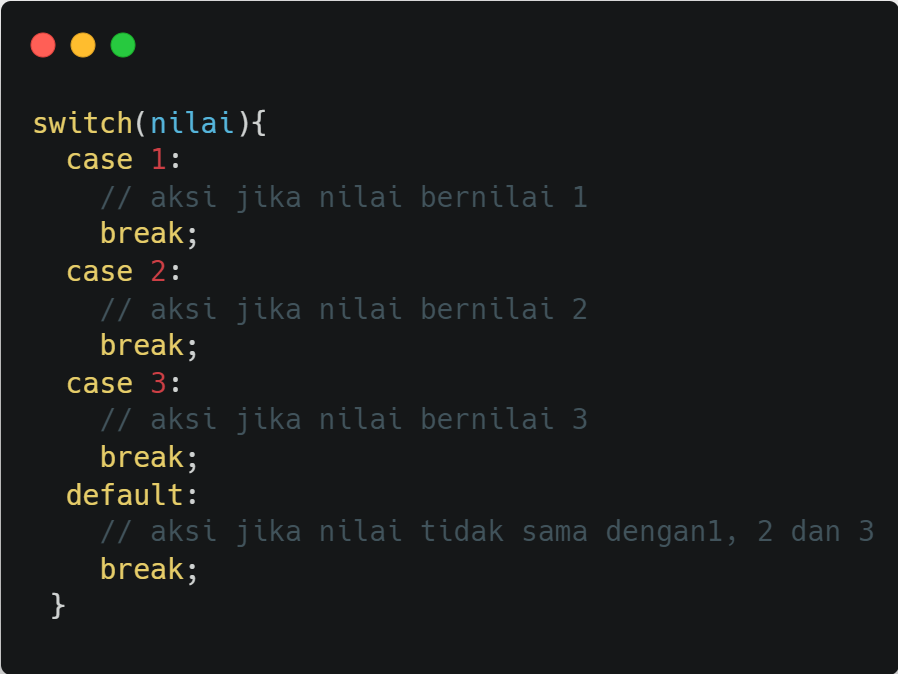


**Gambar 3. Lebih dari 2 Kasus (If – Else if - Else)**

Aksi akan dijalankan jika kondisi1 terpenuhi, jika kondisi1 tidak terpenuhi maka program akan memeriksa pada kondisi2, dan jika kondisi1 dan kondisi2 tidak terpenuhi maka program akan langsung mengeksekusi aksi pada bagian else.

## SWITCH

Jenis pemilihan kedua yaitu switch case yang biasanya digunakan untuk kasus pemilihan yang lebih banyak, dan juga agar baris kode yang dibuat lebih sederhana dan rapi. Switch case ini sering juga digunakan untuk membuat menu-menu pada program yang menggunakan inputan integer maupun character.



**Gambar 4. Contoh Penggunaan Switch Case**

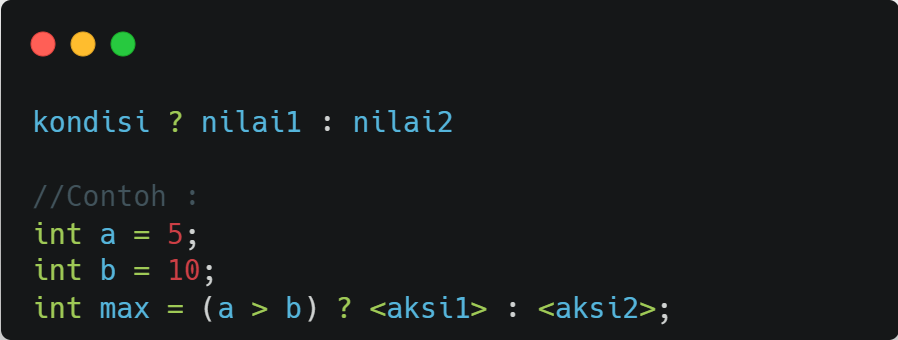
Pada switch, kondisi dari sebuah percabangan terletak setelah ‘case’, pada contoh diatas aksi akan diajalankan jika nilai dari sebuah variable menunjukkan angka 1 atau 2 dan jika kedua nilai itu tidak terpenuhi maka aksi default akan dijalankan, default berarti else. Break pada pemilihan switch berfungsi untuk memaksa program keluar dari suatu kondisi pada pemilihan agar aksi pada kondisi selanjutnya tidak tereksekusi.

## OPERATOR TERNARY

Sebuah gambar berisi teks, cuplikan layar, Font

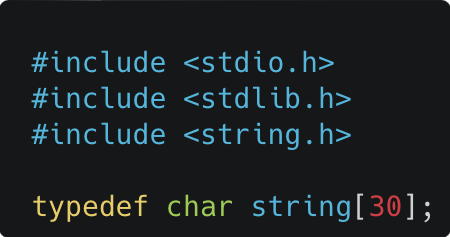
Description automatically generatedOperator ternary (atau conditional operator) adalah operator yang digunakan untuk membuat keputusan dalam satu baris kode. Operator ini memiliki tiga operand, dan digunakan untuk melakukan evaluasi suatu kondisi, kemudian menghasilkan nilai berdasarkan kondisi tersebut.

**Gambar 5. Rumus Operator Terminary**

Jika kondisi bernilai true, maka nilai1 akan dihasilkan, sedangkan jika kondisi bernilai nilai2 akan dihasilkan.

**Gambar 6. Contoh Penggunaan Operator Terminary**

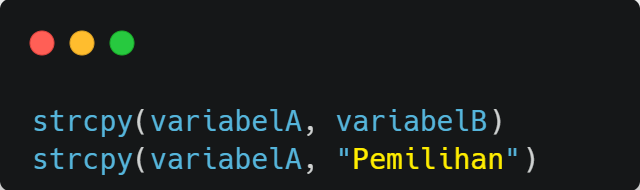
# STRING



**Gambar 7. Library <string.h> dan Tipe Data Bentukan String**

Pada Bahasa pemograman C tipe data string merupakan tipe data bentukan, dimana dibentuk dari tipe data char dengan mendeklarasikan typedef char string[<max karakter>] di header. Pada pemilihan akan erat kaitannya dengan suatu pengkondisian bertipe data string, namun untuk bisa menggunakan fungsi-fungsi yang akan membantu dalam perbandingan didalam pengkondisian ini perlu memanggil library #include <string.h> pada header. Didalam library ini terdapat beberapa fungsi yaitu strcpy, strlen, strcmp dan strcmpi.

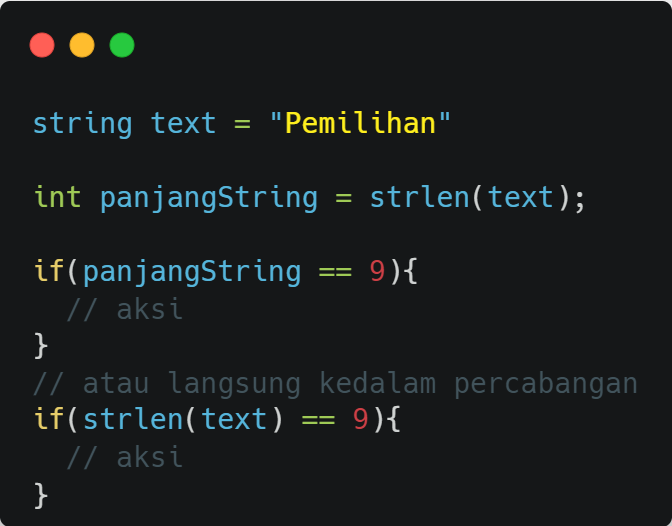
## STRCPY



**Gambar 8. Contoh Penggunaan Strcpy**

Fungsi strcpy merupakan *string copy*, yang berfungsi untuk menyalin sebuah *string* kedalam suatu variabel diruas kirinya. Seperti contoh di atas menyalin nilai dari variabelB kedalam variabelA dan memasukan string “Pemilihan” kedalam variabelA.

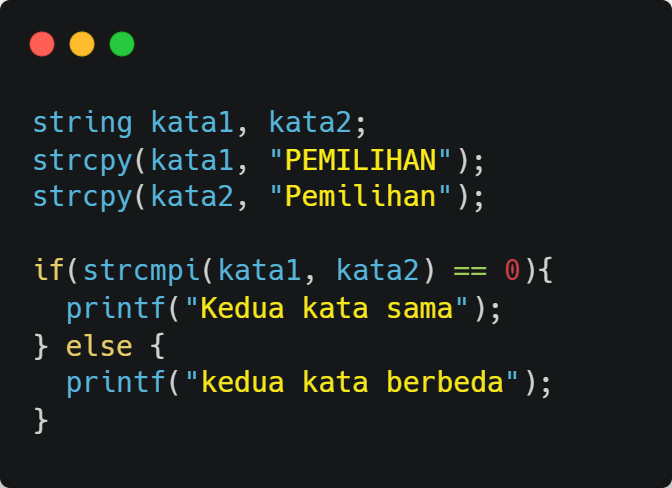
## STRLEN



**Gambar 9. Contoh Penggunaan Strlen**

Fungsi strlen merupakan *string length*, yang berfungsi untuk mengetahui berapa jumlah karakter yang terdapat dalam sebuah *string* atau variabel dengan tipe data *string*. Penggunaan fungsi strlen ini akan jauh lebih efektif jika menempel pada suatu variabel / dijadikan sebagai kondisi pada pemilihan.

## STRCMP & STRCMPI



**Gambar 10. Contoh Penggunaan Strcmp & Strcmpi**

Fungsi strcmp dan strcmpi sama-sama berfungsi untuk membandingkan dua buah *string*. Strcmpi merupakan *string compare insensitive* dimana fungsi ini akan menganggap kedua *string* sama jika semua karakter di variabel kata1 dan kata2 identik, tidak peduli dengan huruf kapital atau tidak. Sedangkan strcmp merupakan *string compare* dengan *case sensitive*, artinya fungsi akan menganggap kedua string sama / *true* jika semua karakter benar-benar identik termasuk memperhatikan huruf kapitalnya. Kedua fungsi ini sering digunakan sebagai kondisi di perulangan. Seperti contoh diatas akan mengeluarkan output “Kedua kata sama” karena menggunakan *case insensitive* yaitu strcmpi. Dan jika menggunakan strcmp maka program akan mengeluarkan output kedua kata berbeda.

**Gambar 11. Contoh Penggunaan Strcmp dan sama dengan**

Akan bernilai *True* jika variabelA dan variabelPembanding memiliki isi *string* yang sama.



**Gambar 12. Contoh Penggunaan Strcmp dan tidak sama dengan**

Akan bernilai *True* jika variabelA dan variabelPembanding memiliki isi string yang berbeda.

# GUIDED

1. Buatlah sebuah program yang terdapat 3 menu, yaitu menu bilangan prima dan menu string compare. Menu bilangan prima akan meminta inputan bilangan bebas, dan mengeluarkan output apakah bilangan yang di inputkan tersebut merupakan bilangan prima atau bukan prima. Menu string compare, berfungsi untuk mengecek apakah kedua inputan string bernilai sama atau tidak. Yang terakhir, menu pengecek nilai akan meminta inputan nama mahasiswa dan nilai mahasiswa, setelah itu akan menampilkan hasil dari pengecekan nilainya dengan ketentuan :

|  |  |
| --- | --- |
| Nilai | Huruf |
| >= 85 | A |
| 65 - 84 | B |
| 40 - 64 | C |
| 20 - 39 | D |
| <= 19 | E |

**\*Note :**

a. Comment tidak perlu di ketikkan juga.

b. Untuk menu 1 & menu 2 sudah terdapat jawaban nya di guided.

c. Buatlah kode jawaban dari menu 3 sesuai dengan permintaan soal.



**Gambar 13. Jawaban Guided**

Penjelasan Kode:

Pada menu 1, program akan meminta inputan bilangan bebas. Jika bilangan yang di inputkan lebih kecil dari sama dengan 1 maka program akan menampilkan bahwa bilangan tersebut bukan lah prima. Selanjutnya jika bilangan lebih besar dari 1, program akan mengecek lagi apakah bilangan adalah 2, 3, 5 atau 7, atau bilangan tidak habis di bagi 2, 3, 5, dan 7, maka jika memenuhi kondisi tersebut, program akan menampilkan bahwa bilangan tersebut adalah prima. Jika tidak memenuhi kondisi tersebut, maka bilangan bukan lah prima.

Pada menu 2, program akan meminta inputan dua string, setelah itu di cek menggunakan fungsi strcmp dan strcmpi. Dan mengeluarkan output jika memenuhi kondisi atau tidak.

Pada menu 3, program akan meminta inputan nama dan nilai mahasiswa, setelah itu program akan mengeluarkan output huruf nilai sesuai dengan ketentuan soal.

Output Kode:

1. Menu 1 :

Sebuah gambar berisi teks, Font, cuplikan layar

Description automatically generated

**Gambar 14. Output dari Guided Menu 1**

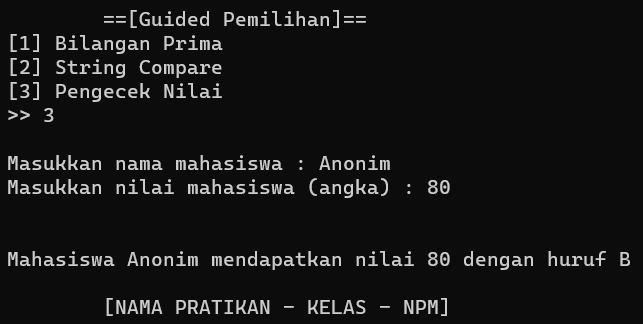
1. Menu 2:

Sebuah gambar berisi teks, cuplikan layar, Font

Description automatically generated

**Gambar 15. Output dari Guided Menu 2**

1. Menu 3:



**Gambar 16. Output dari Guided Menu 3**

# Ketentuan Pengerjaan:

Pastikan ekstensi program file main adalah .c **bukan** .cpp

1. Pastikan semua file dikerjakan dalam satu folder (termasuk file .dev dan .c).
2. Guided dikerjakan dalam folder dengan format **GD4\_X\_YYYYY** lalu di zip.

X = Kelas

YYYYY = 5 digit terakhir NPM

**Kesalahan pengumpulan format penamaan akan dikurangi 10 poin.**

**Harap diperhatikan!**