

Ikhtisar

Condition adalah bagaimana *programmer* dapat membuat keputusan dalam program, dengan memungkinkan beberapa bagian dari kode hanya berjalan dalam keadaan tertentu. *Condition* umumnya akan berfungsi dengan mengevaluasi **Boolean expressions**, yang merupakan ekspresi yang akan memiliki nilai **true** atau **false**. *Programmer* dapat mengatur kondisi sedemikian rupa sehingga kode yang berbeda akan berjalan tergantung pada apa nilai dari *Boolean expression*.

Istilah Kunci

- *condition*
- *Boolean expression*
- *if statement*
- *switch statement*
- *ternary operator*

1	bool a = 3 < 5;	a true
2	bool b = 2 >= 8;	b false
3	bool c = a && b;	c false
4	bool d = a b;	d true
5	bool e = !d;	e false

Boolean Operators

Boolean operators digunakan untuk membuat *Boolean expressions* yang mengevaluasi **true** atau **false**. *Boolean operators* umum termasuk operator perbandingan: **<** (kurang dari), **>** (lebih besar dari), **==** (sama dengan), **<=** (kurang dari atau sama dengan), **>=** (lebih besar dari atau sama dengan), dan **!=** (tidak sebanding dengan). Misalnya, pada baris 1 di kiri, **a** disetel ke **true** karena ekspresi **3 < 5** benar (karena 3 memang kurang dari 5). Pada baris 2, **b** disetel ke **false** karena ekspresi **2 >= 8** tidak benar.

Operator logis juga dapat digunakan untuk menggabungkan *Boolean expressions*. **&&** adalah operator AND logis: ia akan mengevaluasi ke **true** jika kedua ekspresi di kedua sisi itu benar. **||** adalah operator OR logis: ia bernilai **true** jika setidaknya satu dari dua ekspresi di kedua sisi adalah benar. Dan **!**, operator NOT logis, mengevaluasi kebalikan dari ekspresi apa pun setelahnya.

Conditions

Conditional branching mengacu pada gagasan bahwa bagian kode yang berbeda harus dijalankan dalam keadaan yang berbeda. Jenis kondisional yang paling umum adalah **if statement**: di mana blok kode tertentu (yang berada dalam tanda kurung kurawal) hanya akan berjalan jika kondisi (apa pun yang ada dalam tanda kurung setelah kata **if**) mengevaluasi **true**.

Secara opsional, C juga memungkinkan Anda untuk memasukkan blok **else** setelah pernyataan **if**, yang menentukan kode mana yang harus dijalankan jika kondisi **if** bernilai **false**. C juga akan memungkinkan Anda untuk memasukkan satu atau beberapa pernyataan **else if** setelah pernyataan **if**, untuk menambahkan kondisi tambahan yang dapat menjalankan blok kode yang berbeda. Pernyataan **if** di kanan (baris 1-12) akan mencetak "**positive\n**" jika nilai **x** lebih besar dari 0, "**negative\n**" jika nilai **x** kurang dari 0, dan "**zero\n**" jika nilai **x** sama dengan 0.

C juga memiliki cara lain untuk mengekspresikan *conditionals*. **Switch statement**, ditunjukkan di sebelah kanan (baris 15-25), mengambil satu variabel, dan menentukan kode apa yang harus dijalankan berdasarkan **case** yang cocok dengan variabel tersebut. Dalam contoh di sebelah kanan, jika **x** sama dengan 1, "**A\n**" dicetak; jika **x** sama dengan 2, "**B\n**" dicetak, dan dalam semua kasus lainnya (kasus **default**), "**C\n**" dicetak. Kode di dalam **case** harus diakhiri dengan **break** sehingga program tahu kapan untuk berhenti mengeksekusi kode dan pergi ke akhir pernyataan **switch**.

Ternary operator adalah jenis kondisi ketiga. **Ternary operator** mengambil sebuah ekspresi, dan mengevaluasi ke satu nilai jika ekspresi itu **true**, dan nilai lain jika itu **false**. Pada contoh pada baris 28, jika **x > 3**, **y** diatur ke 2, dan 1 sebaliknya.

```

1  if (x > 0)
2  {
3      printf("positive\n");
4  }
5  else if (x < 0)
6  {
7      printf("negative\n");
8  }
9  else
10 {
11     printf("zero\n");
12 }
13
14
15 switch (x)
16 {
17     case 1:
18         printf("A\n");
19         break;
20     case 2:
21         printf("B\n");
22         break;
23     default:
24         printf("C\n");
25 }
26
27
28 int y = (x > 3) ? 2 : 1;
```