Boolean Expressions & Conditionals

Warning: This document is a work in progress.

Variables CSP 3.1

Variable Declaration

You can think of a variables as a box with a label on it. We can look what value is in the box or we can put a new value in the box.

```
x ← 5
```

This declares a variable x and assigns it the value 5.
We can increase the value of x by 1 like this:

```
x ← x + 1
```

We can also work with several variables at once.

```
x ← 5
y ← 10
z ← x + y
```

This will set z to 15.

Boolean Expressions CSP 3.5

Categories

- **Numbers, Strings, Variables**
- **Logical Operators**
- **Relational Operators**
- **Parentheses**

Relational Operators

Relational operators compare two values and evaluate to either true or false.

Code Symbol	AP Symbol	Meaning
`==`	$=$	Equal to
`!=`	$\neq$	Not equal to
`>`	$>$	Greater than
`<`	$<$	Less than
`>=`	$\geq$	Greater than or equal to
`<=`	$\leq$	Less than or equal to

Logical Operators

Logical operators combine two or more boolean expressions and evaluate to either true or false.

Code Symbol	AP Symbol	Meaning
`&&`	AND	Logical AND (true only if both are true)
`||`	OR	Logical OR (true if at least one is true)
`!`	NOT	Logical NOT (true if false, false if true)

Examples

- *Example 1:*

  x > 10 AND y < 20

  True for $x = 11, y = 19$

  False for $x = 11, y = 21$

- *Example 2:*

  NOT (a == b)

  True for $a = 5, b = 2$

  False for $a = 11, b = 11$

Side by Side Comparison

Operation	JavaScript	AP Pseudocode
Assignment	x = 5	x ← 5
Equal to	a == b	a = b
Logical OR	a || b	a OR b
Logical AND	a && b	a AND b
Logical NOT	! a	NOT a

Conditionals (AP CSP 3.6 & 3.7)

Basic IF Statement

This is the basic structure of an if statement:

```
IF (condition)
{
    statements
}
```

If the condition is true, the statements inside the if block are executed.

Here's an example:

```
score ← 100
IF (score ≥ 100)
{
    bonus ← score * 10
}
```

This example will set bonus to 1000 because the condition is true.

IF ELSE Statements

This is the structure of an if else statement:

```
IF (condition)
{
    statements
}
ELSE
{
    statements2
}
```

If the condition is true, the statements inside the if block are executed. If the condition is false, the statements2 inside the else block are executed.

Here's an example:

```
score ← 90
IF (score ≥ 100)
{
    bonus ← score * 10
}
ELSE
{
    bonus ← 0
}
```

This example will set bonus to 0 because the condition is false.

IF ELSE IF (ELSE) Statements

This is the structure of an if else if statement:

```
IF (condition)
{
    statements
}
ELSE IF (condition2)
{
    statements2
}
ELSE
{
    statements3
}
```

If condition is true, statements inside the if block are executed.
If condition is false, condition2 is checked.
If condition2 is true, statements2 inside the else if block are executed.
If condition2 is false, statements3 inside the else block are executed (if present).
We can have multiple else if statements, but only one else statement (or no else statement).

Here's an example:

```
score ← 85
IF (score ≥ 90)
{
    grade ← "A"
}
ELSE IF (score ≥ 80)
{
    grade ← "B"
}
```

This example will set grade to "B".

0.1 Multiple IF Statements

Else connects its block to the previous one to build one unit. Of that whole structure that starts with a single if statement, only one block will be executed.

However, multiple if statements that are not connected with an else statement are executed independently. Here's an example:

```
score ← 100
count ← 0
IF (score ≥ 80)
{
    count ← count + 1
}
IF (score ≥ 90)
{
```

```
9           count ← count + 1
10      }
11      count ← count + 1
12      IF (score ≥ 100)
13      {
14          count ← count + 1
15      }
```

This example will set count to 4 because all three conditions are true and we add 1 to count without a condition between the last two if statements.

Nested Conditionals

We can have if statements inside other if statements.

Here's an example:

```
1       homeworkComplete ← true
2       score ← 95
3       IF (homeworkComplete)
4       {
5           IF (score ≥ 90)
6           {
7               grade ← "A"
8           }
9           ELSE
10          {
11              grade ← "B"
12          }
13      }
```

We check if homeworkComplete is true. Only if it is true, we check if score is greater than or equal to 90. In this case, grade will be set to "A".

The curly brackets { } indicate what statements are inside a block. To see more easily see the blocks, we can use indentation. All statements inside a block usually have the same indentation (like the statements in lines 7 to 12. They are all part of the if(homeworkComplete) block)