



University
of Glasgow

UESTC 1005 – Introductory Programming

Lecture 3 – Operators and Program Control

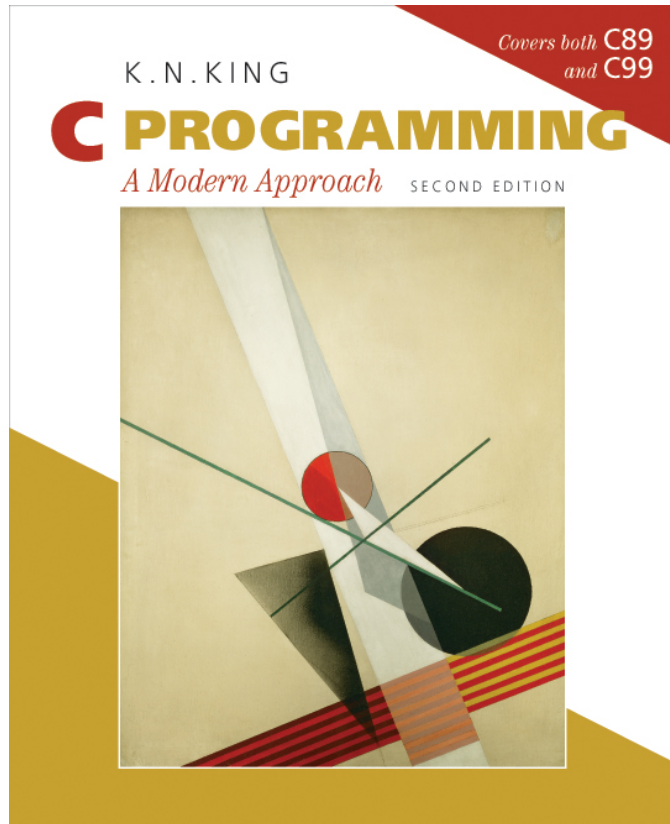
Dr Hasan T Abbas

Hasan.Abbas@glasgow.ac.uk

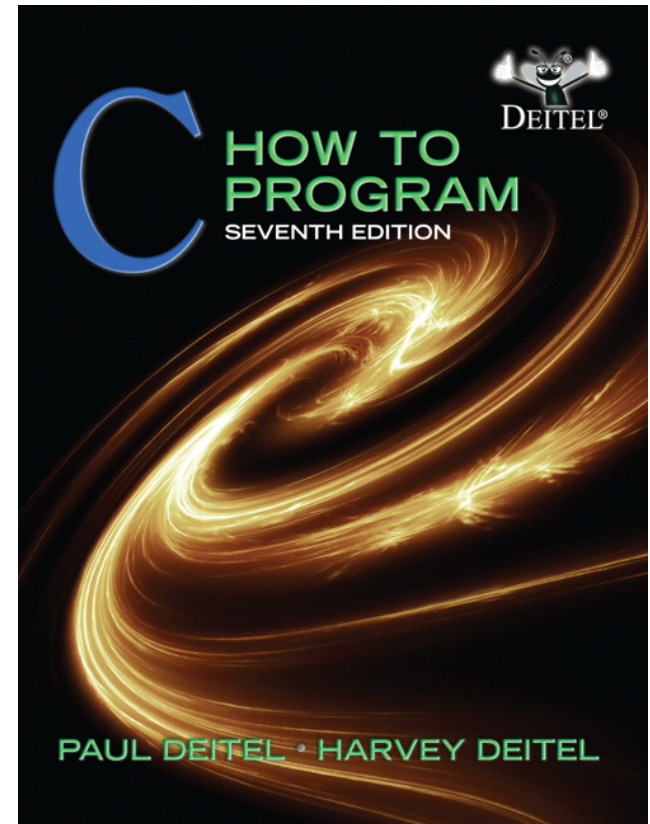
Fall 2019

Glasgow College – UESTC

Suggested Reading for this Week



King – Chapters 5,
6, 9



Detiel – Chapters
3, 4, 5

Relational Operators – Decision Making

Standard algebraic equality operator or relational operator	C equality or relational operator	Example of C condition	Meaning of C condition
<i>Equality Operators</i>			
=	==	x == y	x is equal to y
≠	!=	x != y	x is not equal to y
<i>Relational Operators</i>			
>	>	x > y	x is greater than y
<	<	x < y	x is less than y
>=	>=	x >= y	x is greater than or equal to y
<=	<=	x <= y	x is less than or equal to y

Logical Operators – Decision Making

Logic AND operator (&&)

Only TRUE when both the expressions are TRUE at the same time

expression1	expression2	expression1 && expression2
0	0	0
0	nonzero	0
nonzero	0	0
nonzero	nonzero	1

Logical Operators – Decision Making

Logic OR operator (||)

TRUE when either of the expressions is TRUE

expression1	expression2	expression1 expression2
0	0	0
0	nonzero	1
nonzero	0	1
nonzero	nonzero	1

Logical Operators – Decision Making

Logic NOT operator (!)

Reverses the outcome of a condition

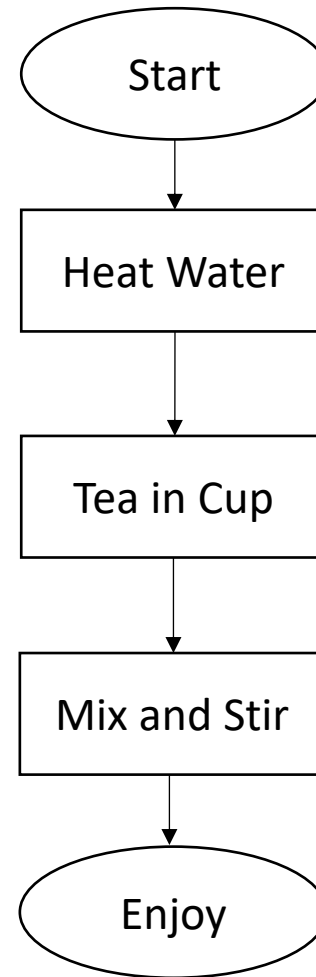
expression	!expression
0	1
nonzero	0

Algorithms

- We solve a computing problem using a series of actions, in a particular order.
 - There is a procedure that we need to follow
 - Series of actions
 - In a particular order
 - We call the procedure an algorithm
-
- EXAMPLE – Make a cup of tea
 1. Heat up the water
 2. Put tea in a cup
 3. Pour hot water in the cup
 4. Mix and stir
 5. Enjoy!

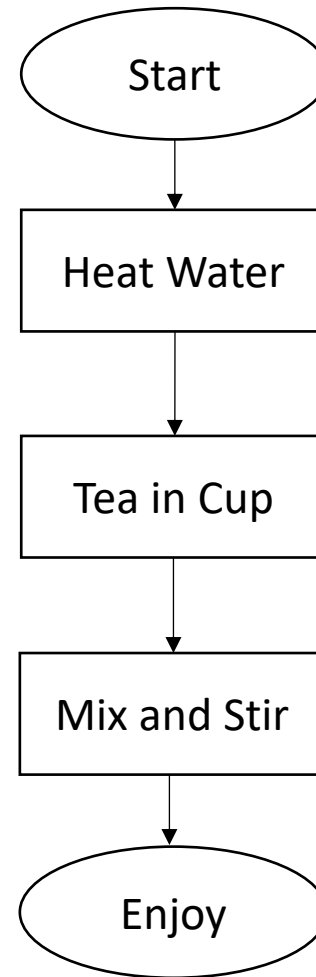
Flowcharts

- Flowcharts are a graphical representation of an algorithm
- EXAMPLE – Make a cup of tea
 1. Heat up the water
 2. Put tea in a cup
 3. Pour hot water in the cup
 4. Mix and stir
 5. Enjoy!

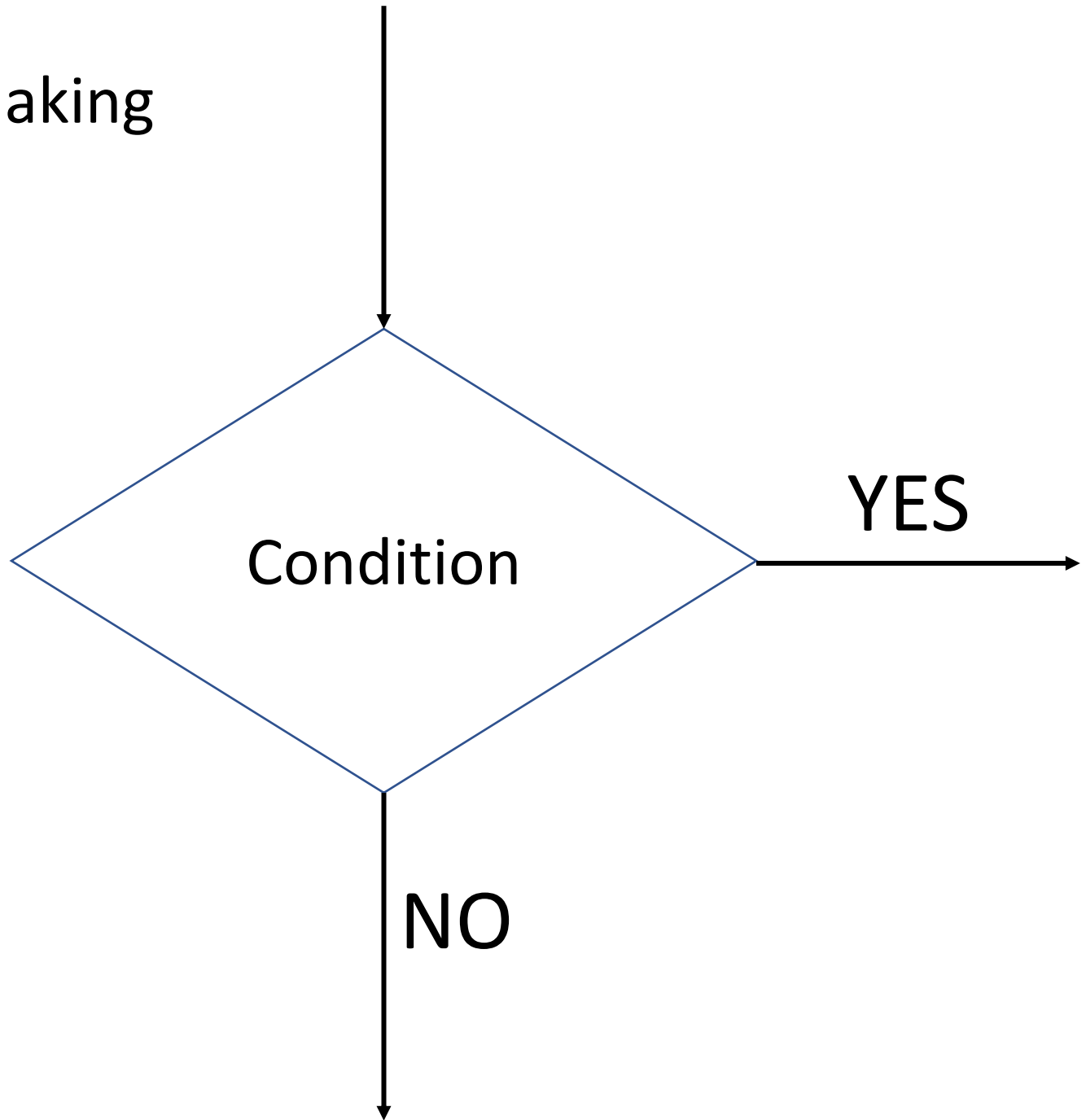


Flowcharts

- An oval represents the start or end of the flowchart
- A rectangle is used to represent an action
- A diamond is used for a decision
- Arrows determine the flow of a program



Decision Making



Condition

YES

NO

Pseudocode

- An informal language that helps us develop algorithms
- Written in simple English
- Not executed or compiled
- A simple outline of the computer program

```
If student's grade is greater than or equal to 60
    Print "passed"
else
    Print "failed"
endif
```

Program Control in C programming

C has a lot of operators, but not many statements. So far, we have seen:

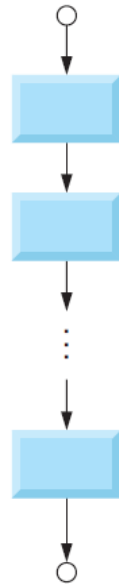
- Expression statements (`a = b + c;`)
- Return statement (`return 0;`)

Now we will see:

- Selection statements
 - `if` statement
 - `if-else` statement
 - `switch` statement
- Iteration statements
 - `for` loop
 - `while` loop
 - `do-while` loop
- Jump statements
 - `break`
 - `continue`

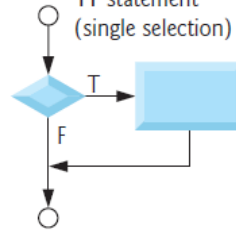
Applying Conditions

Sequence

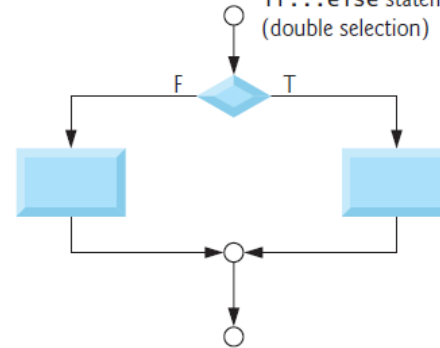


Selection

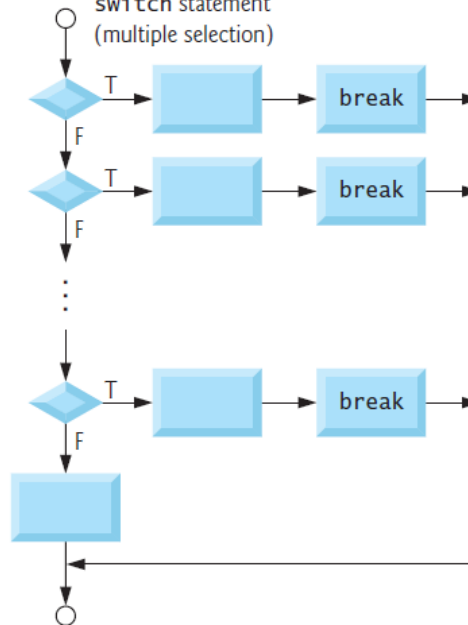
if statement
(single selection)



if...else statement
(double selection)



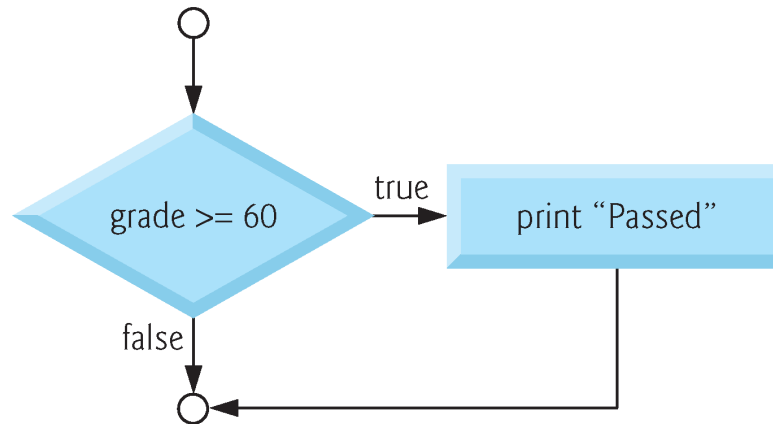
switch statement
(multiple selection)



The if selection statement

The if statement allows a program to select a particular path of execution from a set of options.

```
if (logical expression){  
    expression statements;  
}
```



Example

Applying logic is one of the most important features of programming languages

What happens to my grades **IF** I complete all the labs.

Erin will only go to party **IF** Qin is going.

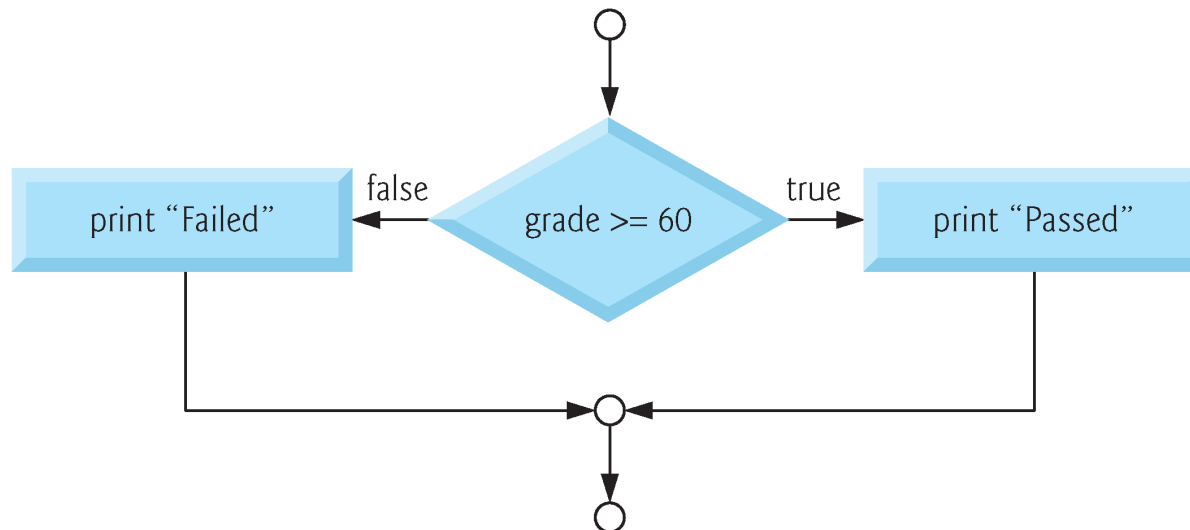
IF it is warm outside, THEN turn on the AC, **ELSE** keep it off.

```
int main(){  
  
    int thermostat = 25; // warm temperature ☹️  
    int room_temperature = 30;  
    int ac_switch; // ON -> 1; OFF -> 0  
    if (room_temperature > thermostat)  
    {  
        ac_switch = 1; // Turn ON the air conditioner  
        printf("It is HOT! ;-0 \n\n TURNING ON THE AC");  
    }  
    return 0;  
}
```

The if-else selection statement

We can specify different actions when the condition is either true or false

```
if (logical expression){  
    expression statements;  
}  
else {  
    expression statements;  
}
```



Example (if-else)

What happens to my grades **IF** I complete all the labs.

What happens otherwise (**ELSE**)

```
int main(){  
    if (grade >= 60)  
    {  
        printf("You passed ;-)\n");  
    }  
    else  
        printf("You failed :-(\n");  
    return 0;  
}
```

The if-else if – else selection statement

We can specify multiple actions when for multiple conditions

```
if (logical expression){  
    expression statements;  
}  
else if(logical expression) {  
    expression statements;  
}  
else  
    expression statements;
```

Example (if-else)

What happens to my grades **IF** I complete all the labs.

What happens otherwise (**ELSE**)

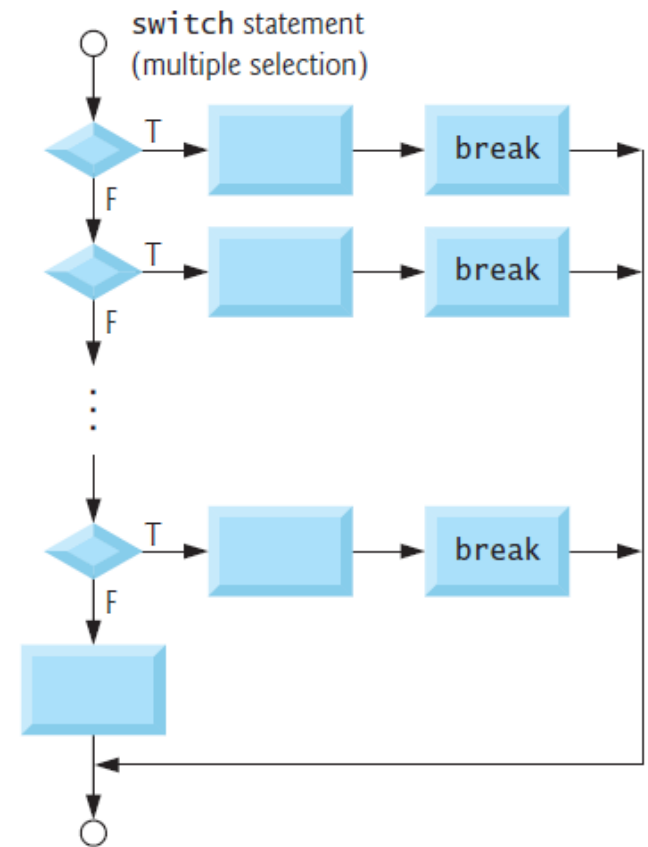
```
int main(){  
    if (grade >= 90)  
    {  
        printf("You passed with A grade ;-)\n");  
    }  
    else if (grade >= 80)  
        printf("You passed with B grade :-)\n");  
    else if (grade >= 70)  
        printf("You passed with C grade :-| \n");  
    else if (grade >= 60)  
        printf("You passed with D grade :-\ \n");  
    else  
        printf("You failed :-( \n");  
    return 0;  
}
```

The switch selection statement

Multiple selection statement; just like multiple choice questions

Switch selection is preferred when the logical expressions are independent of each other

```
switch (variable){  
    case 1:  
        statements;  
        break; // exit switch  
    case 2:  
        statements;  
        break; // exit switch  
    default:  
        statements;  
        break; // exit switch  
}
```



The `break` jump statement

- The `break` statement is necessary to exit the `switch` statement once the particular option has been executed
- Takes the program control outside the `switch` statement
- If `not` used, remaining cases are also executed.

Example (switch)

```
int main(){
    char grade = 'B';
    switch(grade) {
        case 'A' :
            printf("Excellent!\n" );
            break;
        case 'B' :
        case 'C' :
            printf("Well done\n" );
            break;
        case 'D' :
            printf("You passed\n" );
            break;
        case 'F' :
            printf("Better try again\n" );
            break;
        default :
            printf("Invalid grade\n" );
    }
    return 0;
}
```

Short Quiz

ROOM NAME: **AWJUQZW**



<https://b.socrative.com/login/student/>

Next Lecture ...

Loops

`while()` loop

`for()` loop

Functions