COMP133: **COMPUTER AND PROGRAMMING**

Selection Structure

Dr. Radi Jarrar
Department of Computer Science
Birzeit University

Selection Structure

- Selection structure is used to make decision of executing commands.
- It allows to execute a command (or block of commands) based on some conditions, or selecting an alternative course of action if the condition is false.
- Uses `if`, `if…else`, or nested `if…else`.

LOGICAL AND RELATIONAL OPERATORS

Relational & Equality Operators

Operator	Meaning	Type
<	Less than	Relational
<=	Less than or equal	Relational
>	Greater than	Relational
>=	Greater than or equal	Relational
==	Equals	Equality
! =	Not equal	Equality

Logical Operators

- Three types

Operator	**Meaning**
&&	And
||	Or
!	Negation (not)

Operator Precedence

Operator	Precedence
!, +, -, & (unary operators)	Highest
*, /, %	↓
+, -	
<, <=, >, >=	
==, !=	
&&	
||	
=	Lowest

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- `float x=3.0, y=4.0, z=2.0;`
- `int flag = 0;`
- What is the value after applying the following expression:
- `! flag`
- `x + y / z <= 3.5`
- `! flag || (y + z >= x -z )`
- `!(flag||(y + z >= x -z ))`

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

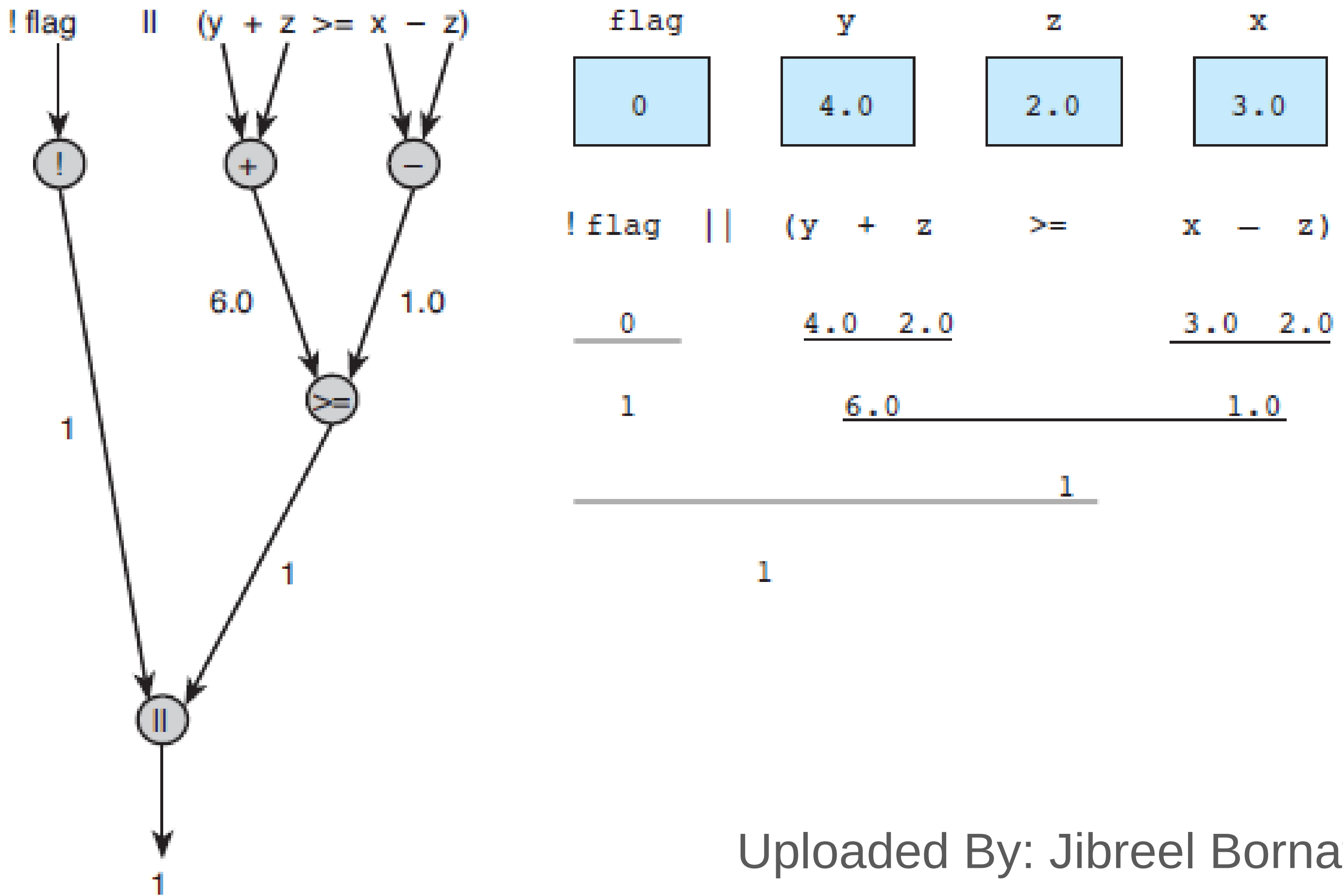
Example

- `float x=3.0, y=4.0, z=2.0;`
- `int flag = 0;`
- What is the value after applying the following expression:
- `! flag` →!0 is 1 (true)
- `x + y / z <= 3.5` →5.0 <= 3.5 is 0 (false)
- `! flag || (y + z >= x -z )` →1 || 1 is 1 (true)
- `!(flag||(y + z >= x -z ))` →!(0 || 1) is 0 (false)

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- Evaluation: `!flag || (y + z  >=  x -z)`



From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- How to translate conditions into C-code?

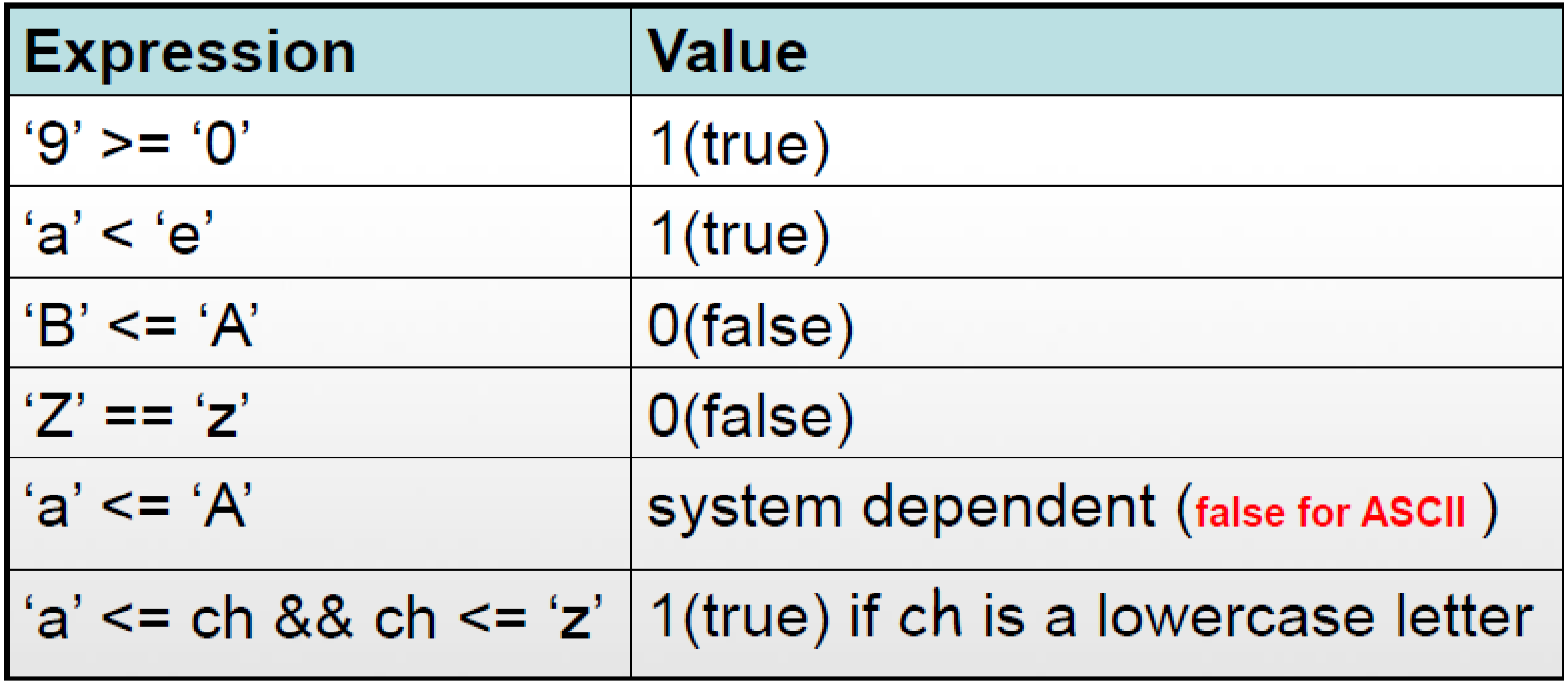
```
int x = 3, y = 4, z = 2;
```

Condition (English)	Logical Expression	Evaluation
x and y are greater than z	`x > z && y > z`	`1 && 1 is 1 (true)`
x is equal to 1 or 3	`x==1 || x==3`	`0 || 1 is 1 (true)`
x is in the range z to y, inclusive	`z<=x && x<=y`	`1 && 1 is 1 (true)`

x is outside the range z to y	`!(z<=x && x<=y)`
	`OR`
	`z>x || x>y`
		`!(1 && 1) is 0 (false)`
		`0 || 0 is 0 (false)`

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Comparing Characters



Expression	Value
‘9’ >= ‘0’	1(true)
‘a’ < ‘e’	1(true)
‘B’ <= ‘A’	0(false)
‘Z’ == ‘z’	0(false)
‘a’ <= ‘A’	system dependent (false for ASCII)
‘a’ <= ch && ch <= ‘z’	1(true) if ch is a lowercase letter

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Logical Assignment

```
#include<stdio.h>
int main(){
     int grade, hasPassed;

     printf(“Please enter a grade”);
     scanf(“%d”, &grade);

     hasPassed = ( grade >= 60 );
     printf(“The student passed the course %d”, hasPassed);
     return 0;
}
```

IF STATEMENT

If Statement - Structure

```
if( condition )
{
    //statements when true
}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

If Statement – Strcutre with Compound Statements

```
if( condition )
{
    //statements when true
}
else
{
    //statements when false
}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

If Statements

- If statement with **one alternative** (i.e., without else)

```
if(x!=0)
    product = product * x;
```

- If statement with **two alternatives**

```
if (rest_heart_rate>56)
    printf(“Your heart is in excellent
health!\n”);
else
    printf(“Keep up your exercise program!\n”);
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- Write a program that prints if a student has passed/failed a course (given the pass is 60 and above).

Example

```
#include<stdio.h>
int main(){
    int grade;
    printf(“Please enter a grade”);
    scanf(“%d”, &grade);

    if( grade >= 60 )
        printf(“Pass\n”);
    else
        printf(“Fail\n”);
    return 0;
}
```

Example

- Write a program that checks if a given number is even or odd.

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

```
#include<stdio.h>
int main(){
     int number;
     printf(“Please enter a number”);
     scanf(“%d”, &number);

     if( number % 2 == 0)
          printf(“Even\n”);
     else
          printf(“Odd\n”);
     return 0;
}
```

Example

- Write a program that reads two values x & y. If x is less than y, then switch their values.

Example

```
#include<stdio.h>
int main(){
     int x, y, temp;
     scanf(“%d%d”, &x, &y);

     if( x < y )
     {
          temp = x;
          x = y;
          y = temp;
     }
     return 0;
```

Nested if

- An if statement with another if statement as its true task or its false task.
- E.g., write an algorithm that checks if a number is positive, negative, or zero

Nested if

-
  ```
  if (exam >= 80)
          if (project >= 90)
                  grade = 'A';
  ```
- Which can be written using the And operator

```
if (exam >= 80 && project >= 90)
        grade = 'A';
```

Nested if-else

- An if statement with another if statement as its true task or its false task.
- E.g., write an algorithm that checks if a number is positive, negative, or zero

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

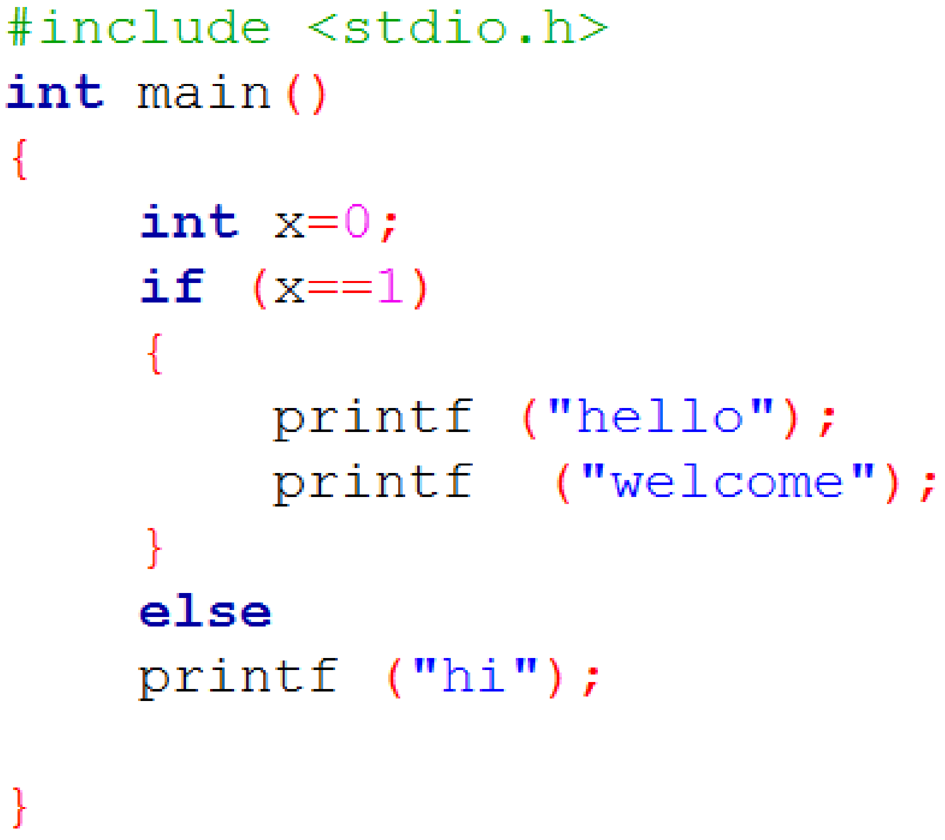
Example

```
#include<stdio.h>
int main(){
      int number;
      printf(“Please enter a number”);
      scanf(“%d”, &number);

      if( number > 0)
            printf(“Positive\n”);
      else if( number < 0 )
            printf(“Negative\n”);
      else
            printf(“Zero”);
      return 0;
}
```

Example

- What is the output of the following program?



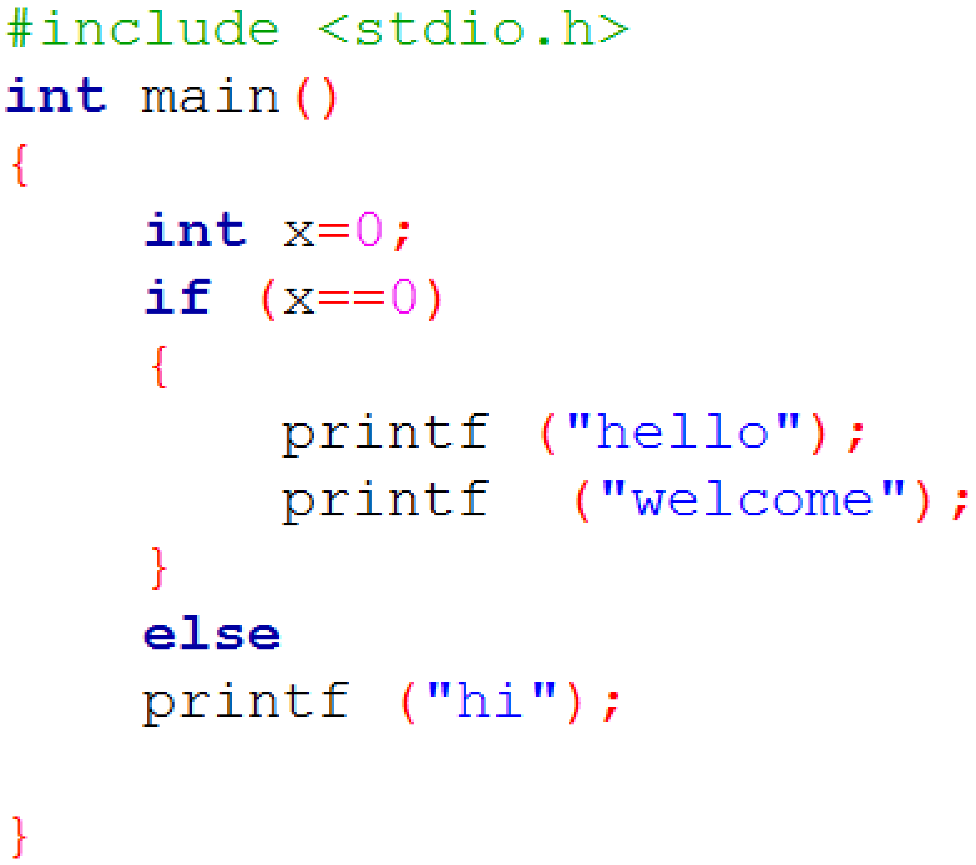
```
#include <stdio.h>
int main()
{
    int x=0;
    if (x==1)
    {
        printf ("hello");
        printf  ("welcome");
    }
    else
    printf ("hi");

}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- What is the output of the following program?



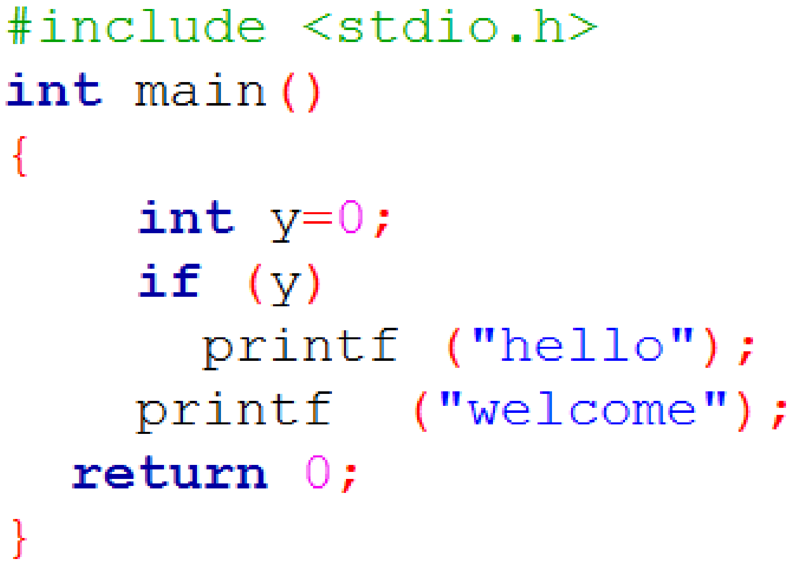
```
#include <stdio.h>
int main()
{
    int x=0;
    if (x==0)
    {
        printf ("hello");
        printf  ("welcome");
    }
    else
    printf ("hi");

}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- What is the output of the following program?

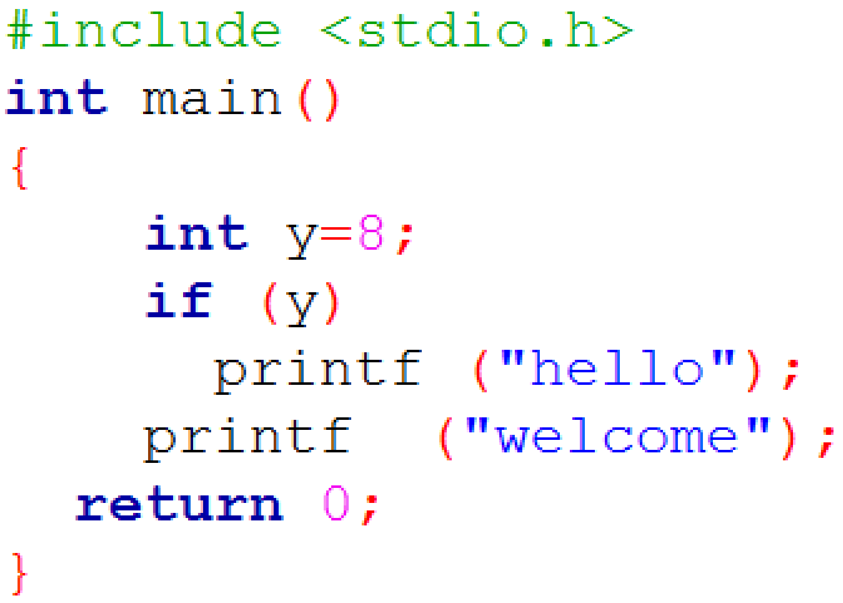


```
#include <stdio.h>
int main()
{
    int y=0;
    if (y)
      printf ("hello");
    printf  ("welcome");
  return 0;
}
```

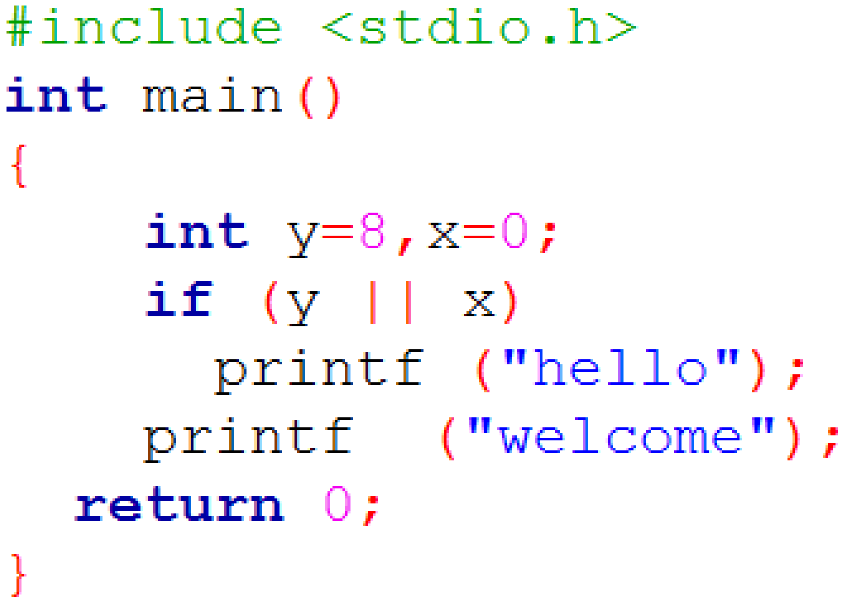
From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- What is the output of the following program?



```c
#include <stdio.h>
int main()
{
    int y=8;
    if (y)
      printf ("hello");
    printf  ("welcome");
  return 0;
}
```

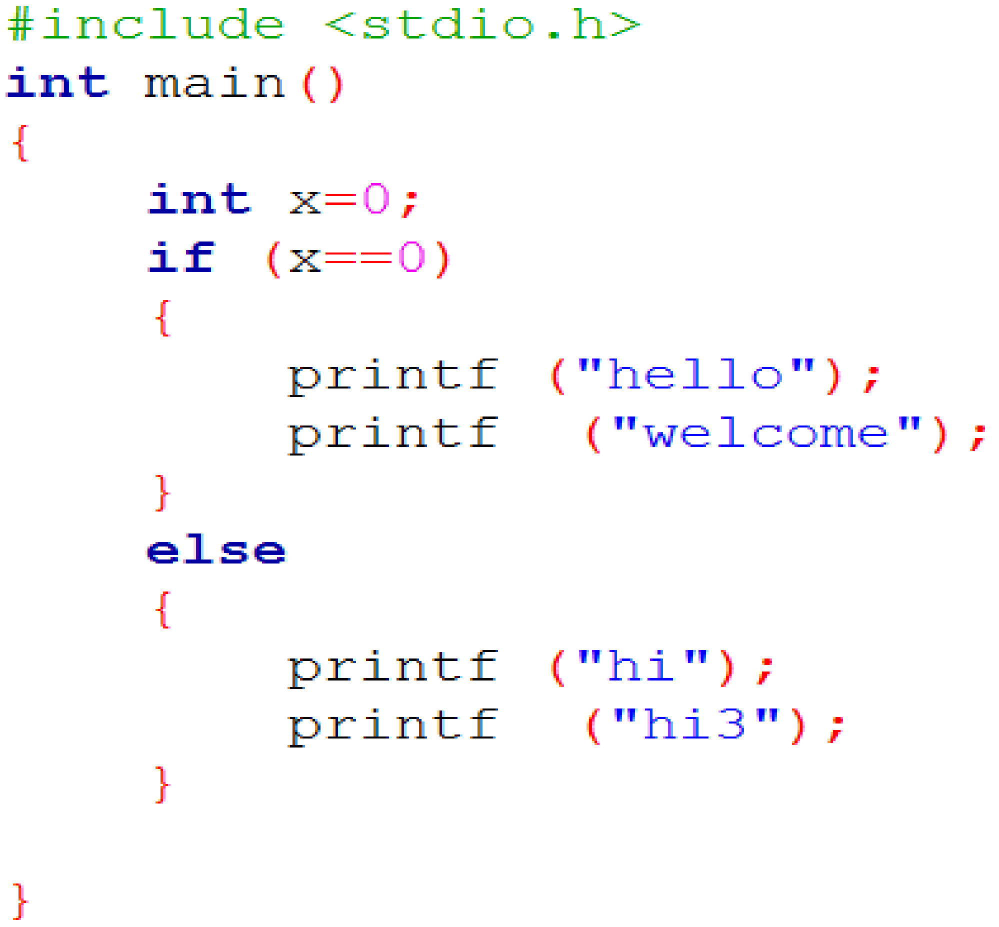


```c
#include <stdio.h>
int main()
{
    int y=8,x=0;
    if (y || x)
      printf ("hello");
    printf  ("welcome");
  return 0;
}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Example

- What is the output of the following program?



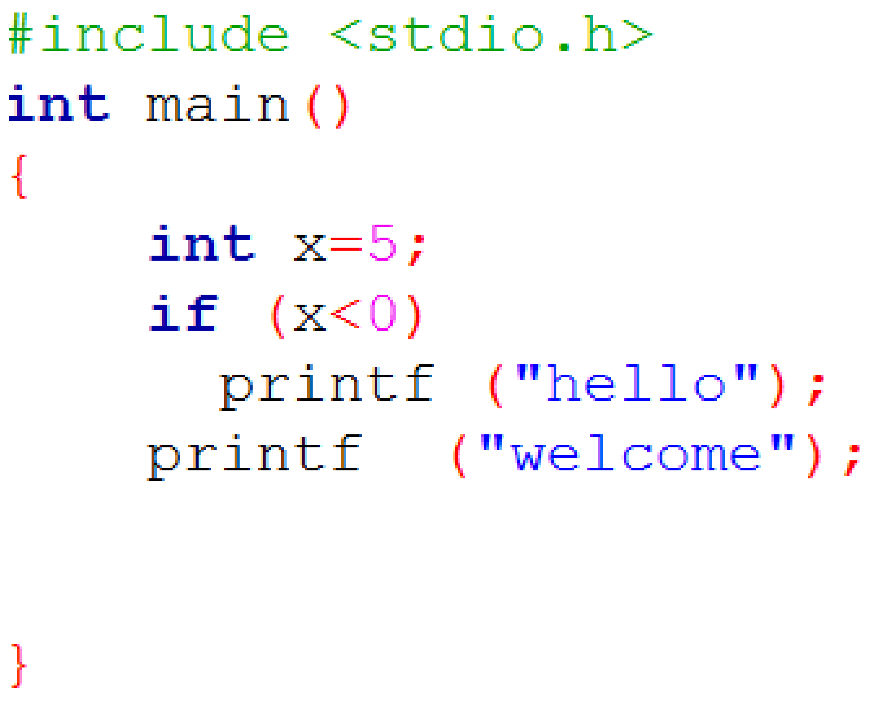
```
#include <stdio.h>
int main()
{
    int x=0;
    if (x==0)
    {
        printf ("hello");
        printf  ("welcome");
    }
    else
    {
        printf ("hi");
        printf  ("hi3");
    }

}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

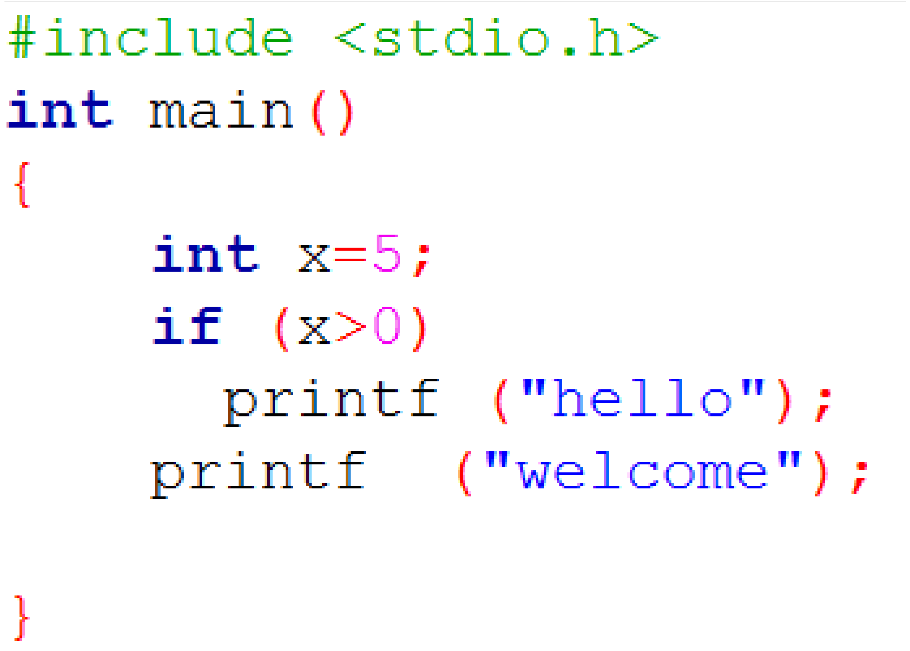
Example

- What is the output of the following program?



```
#include <stdio.h>
int main()
{
    int x=5;
    if (x<0)
      printf ("hello");
    printf  ("welcome");


}
```

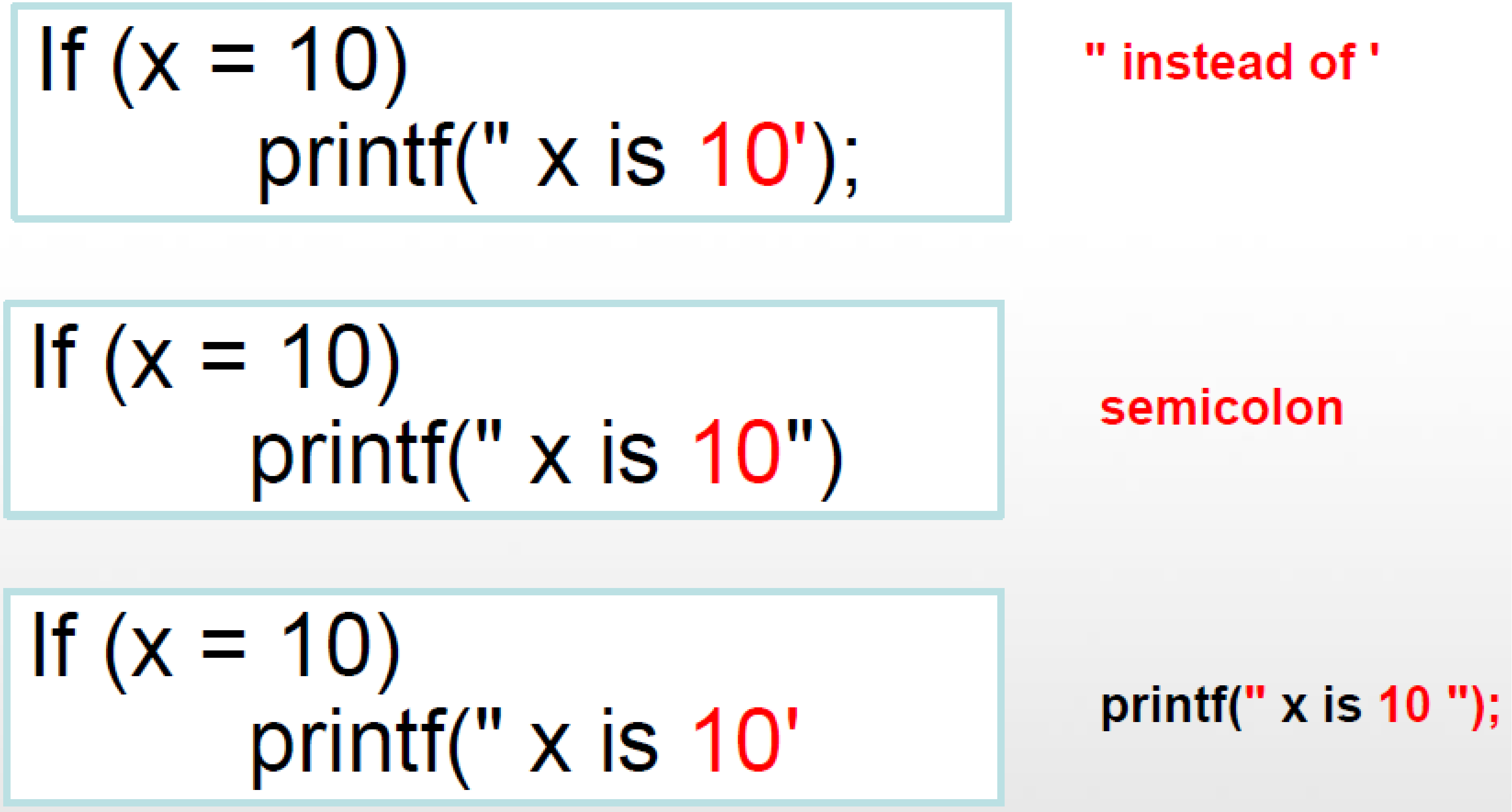


```
#include <stdio.h>
int main()
{
    int x=5;
    if (x>0)
      printf ("hello");
    printf  ("welcome");

}
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Common Errors with if…else



```
If (x = 10)
        printf(" x is 10');
```

" instead of '

```
If (x = 10)
        printf(" x is 10")
```

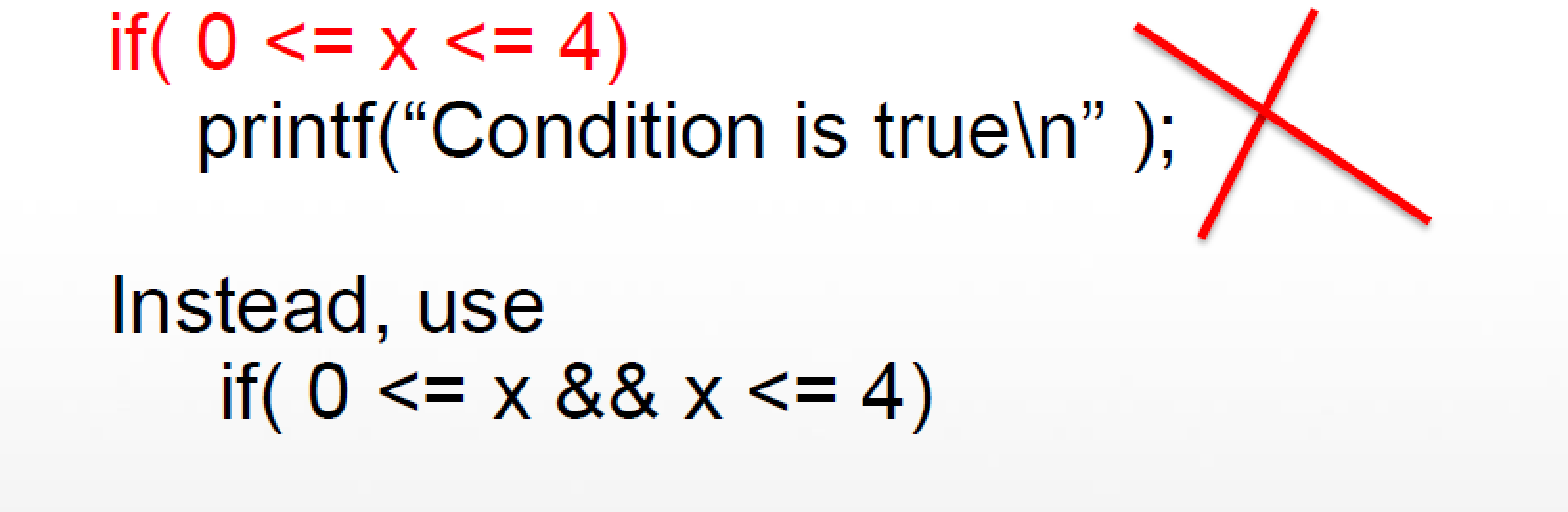
semicolon

```
If (x = 10)
        printf(" x is 10'
```

printf(" x is 10 ");

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

Common Errors with if…else



```
if( 0 <= x <= 4)
    printf(“Condition is true\n” );
```

Instead, use

```
    if( 0 <= x && x <= 4)
```

From the material of Mr. Abdallah Karakra, 2016, Birzeit University.

SWITCH STATEMENT

Switch statement

- The switch statement tests a variable against equality for some values
- It is used only to check for equality
- It tests only for variables of types `int` or `char` only

Switch statement - structure

```
switch( expression ){
      case 1:  statements1;
               break;
      case 2:  statements2;
               break;
      case 3:  statements3;
               break;
      .
      .
      .
      default: statements3;
               break;
}
```

Switch statement

- Having evaluated the expression, control jumps to the appropriate `case` label and its statements are executed.
- The `break;` statement is used to indicate that the statements for this case have finished. Once break is executed, the switch statement exists.
- If there is no `break;`, the execution will fall through to the next statement in the succeeding case.
- There may be at most one default label in a '`switch`'. The purpose of the default case is to capture cases that are not included in the cases as the last `else` in an `if`…`else` statements.

Switch statement - Example

- Read the day number and print the day week. Start the week on Sunday.

Switch statement - Example

```
#include <stdio.h>
int main(){
           int day;
           printf(“Enter a day from 1-7\n”);
           scanf(“%d”, &day);

           switch( day ){
                      case 1: printf(“The day is Sunday\n”);
                                  break;
                      case 2: printf(“The day is Monday\n”);
                                  break;
                      case 3: printf(“The day is Tuesday\n”);
                                  break;
                      case 4: printf(“The day is Wednesday\n”);
                                  break;
                      case 5: printf(“The day is Thursday\n”);
                                  break;
                      case 6: printf(“The day is Friday\n”);
                                  break;
                      case 7: printf(“The day is Saturday\n”);
                                  break;
                      default:
                                 printf(“Invalid day\n”);
                                 break;
                      }
           return 0;
}
```

Switch statement - Example

What is the output of this switch statement given that the day = 4?

```
switch( day ){
        case 1: printf(“The day is Sunday\n”);
                  break;
        case 2: printf(“The day is Monday\n”);
                  break;
        case 3: printf(“The day is Tuesday\n”);
                  break;
        case 4: printf(“The day is Wednesday\n”);
                  break;
        case 5: printf(“The day is Thursday\n”);
                  break;
        case 6: printf(“The day is Friday\n”);
                  break;
        case 7: printf(“The day is Saturday\n”);
                  break;
        default:
                 printf(“Invalid day\n”);
                 break;
        }
```

Switch statement - Example

What is the output of this switch statement given that the day = 4?

```
switch( day ){
        case 1: printf(“The day is Sunday\n”);
        case 2: printf(“The day is Monday\n”);
        case 3: printf(“The day is Tuesday\n”);
        case 4: printf(“The day is Wednesday\n”);
        case 5: printf(“The day is Thursday\n”);
        case 6: printf(“The day is Friday\n”);
                  break;
        case 7: printf(“The day is Saturday\n”);
                  break;
        default:
                 printf(“Invalid day\n”);
                 break;
        }
```

Switch statement - Example

What is the output of this switch statement given that the day = 4?

```
switch( day ){
        case 1: printf(“The day is Sunday\n”);
        case 2: printf(“The day is Monday\n”);
        case 3: printf(“The day is Tuesday\n”);
        case 4: printf(“The day is Wednesday\n”);
        case 5: printf(“The day is Thursday\n”);
        case 6: printf(“The day is Friday\n”);
        case 7: printf(“The day is Saturday\n”);
        default:
                printf(“Invalid day\n”);
        }
```

Switch statement - Example

- Enter a character and check if it is a vowel or consonant.

Switch statement - Example

```
#include <stdio.h>
int main(){
          char letter;
          printf(“Enter a letter\n”);
          scanf(“%d”, & letter);

          switch( letter ){
                    case ‘a’:
                    case ‘A’:
                    case ‘e’:
                    case ‘E’:
                    case ‘i’:
                    case ‘I’:
                    case ‘o’:
                    case ‘O’:
                    case ‘u’:
                    case ‘U’:
                              printf(“Vowel\n”);
                              break;

          default:
                              printf(“Consonant\n”);
                              break;
          }
          return 0;
}
```