Féidearthachtaí as Cuimse
Infinite Possibilities

Week 4 - Tutorial

OLLSCOIL TEICNEOLAÍOCHTA
BHAILE ÁTHA CLIATH
TECHNOLOGICAL
UNIVERSITY DUBLIN

Programming - Week 4

Overview

- Revision – scanf()
- if statement
- if / Else statement
- switch statement
- Programming Pitfalls

Revision: scanf()

- When using the scanf() function, the ampersand (&) is used to pass the address of a variable to the scanf function.
- This is necessary because scanf() needs to modify the value of the variable where the input will be stored
- To to this it needs the memory address of the variable.
- Adding the & to the variable name lets us reference the memory where the variable stores its data.

scanf – Example 1

- Ask a user to input their age. Then display their age.

Example_0.c > ...

```c
#include <stdio.h>

int main(void)
{
    int age = 0;

    printf("Please enter your age: \n");

    scanf("%d", &age);

    printf("You are %d years old\n", age);

    return 0;
}
```

Operators

Operator	Meaning
==	Equivalent to / Equal to
!=	Not equivalent to / Not equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to

Flow of Control

- The flow of control in our programs is used break up the flow of execution using decision making, looping, and branching to execute particular blocks of code.
- This allows us to use logic to offer functionality in our programs.
- We can use control flow to make decisions based on different values and conditions.
- We will explore this further this tutorial.

Conditional Statements

- If Statement
- If / Else
- Switch Statement

Conditional statements

- A **conditional statement** in C is used to perform different actions or computations based on whether a certain condition or set of conditions is true or false.
- This allows the program to **make decisions** and execute code accordingly.
- Conditional statements are fundamental in **controlling the flow** of a program.

Conditions

- A condition will be either true or false
- int age = 10;

What are the values of these conditions?

- Bruce Wayne is Batman
- age > 18
- age < 12
- age > 10
- age >= 10
- 1 == 1

if Statement

- The **if statement** checks a condition and will execute a block of code only if the condition evaluates to true.

If the condition evaluates as true the code block will run

Syntax

```
if (condition) {
    // This code will be executed if condition is true
}
```

Advice:
Always use { and } braces around the statements inside the if statement.

Example_1.c

```c
#include <stdio.h>

int main() {
    int age = 18;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    if ( age >= 18 ) {
        printf("Welcome to the Roxbury\n");
    }

}

```

Ask user to enter their age

OLLSCOIL TEICNEOLAÍOCHTA BHAILE ÁTHA CLIATH
TU DUBLIN
TECHNOLOGICAL UNIVERSITY DUBLIN

Program description:

- A bouncer at a nightclub must ask people their age when entering the niteclub.
- The person will give their age
- The program will give a welcome message if the person is 18 or over

C Example_2.c > ...

```c
#include <stdio.h>

int main() {
    int age;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    printf("Hello, how old are you?:\n");
    scanf("%d", &age);

    if ( age >= 18 ) {
        printf("Welcome to the Roxbury\n");
    }

}

```

Feedback (niteclub owner)

- The niteclub owner loves the new program.
- She has asked if we can say **“sorry, you are too young”** if the person is less than 18
- Can we modify the program to add this functionality

Example_2.c > ...

```c
#include <stdio.h>

int main() {
    int age;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    printf("Hello, how old are you?:\n");
    scanf("%d", &age);

    if ( age >= 18 ) {
        printf("Welcome to the Roxbury\n");
    }

}
```

We need to update the above program….

Adding another if statement

- Based on the new program requirement we can easily add another if statement to the program

Example_2.c > ...

```c
#include <stdio.h>

int main() {
    int age;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    printf("Hello, how old are you?:\n");
    scanf("%d", &age);

    if ( age >= 18 ) {
        printf("Welcome to the Roxbury\n");
    }

    if ( age < 18 ) {
        printf("Sorry you can't enter, you are too young\n");
    }

}
```

If / Else

- Instead of using 2 if statements we could use an if .. else block

Syntax:

```
if (condition)
{
    // execute if condition is true
} else {
    // execute if condition is false
}
```

Example_2a.c > ...

```c
#include <stdio.h>

int main() {
    int age;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    printf("Hello, how old are you?:\n");
    scanf("%d", &age);

    if ( age >= 18 ) {
        printf("Welcome to the Roxbury\n");
    } else {
        printf("Sorry you can't enter, you are too young\n");
    }

    return 0;
}
```

Logical Operators

- The C programming language provides logical operators to use with if / if .. else statements

Logical Operator	Meaning
&&	AND
||	OR
!	NOT

Feedback 2 (niteclub owner)

- The niteclub owner loves the updated program.
- She has asked if we can cater for invites for a private function. The person must be 18 or over **and** have an invite to enter the niteclub.
- Can we modify the program to add this functionality.

Example_2a.c > ...

```c
#include <stdio.h>

int main() {
    int age;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    printf("Hello, how old are you?:\n");
    scanf("%d", &age);

    if ( age >= 18 ) {
        printf("Welcome to the Roxbury\n");
    } else {
        printf("Sorry you can't enter, you are too young\n");
    }

    return 0;
}
```

Logical Operators

- The condition has been updated to check if the person has an invite to use && logical operator. For the updated condition to evaluate as true the person must be 18 or over AND they must have an invite.

Example_3.c > ...

```c
#include <stdio.h>

int main() {
    int age;
    char have_invite;

    printf("Niteclub Example - The Roxbury\n");
    printf("Rules: only people 18 or over can enter....\n");

    printf("Hello, how old are you?:\n");
    scanf("%d", &age);

    printf("Do you have an invite (y or n):\n");
    scanf(" %c", &have_invite);

    if ( age >= 18 && have_invite == 'y' ) {
        printf("Welcome to the Roxbury\n");
    } else {
        printf("Sorry you can't enter, you are too young or have no invite\n");
    }

    return 0;
}
```

Else if ()

- This allows you to test multiple conditions sequentially.
- If one condition is true, the corresponding block of code is executed, and the rest are ignored.
- If none of the conditions are true, the else block (if present) is executed.
- The above code is not very efficient given that it has many if .. else blocks.
- This uses up CPU cycles and is expensive in processing time.
- The **switch statement** can be used as an alternative, which is much more efficient.

Example_4.c > ...

```c
#include <stdio.h>

int main() {
    char grade = ' ';

    printf("Enter a grade: \n");
    scanf("%c", &grade);

    if (grade == 'A')
    {
        printf("Highest grade\n ");
    }
    else if (grade == 'B')
    {
        printf("2nd Highest grade\n "); }
    else if (grade == 'C') {
        printf("3rd Highest grade\n ");
    }
    else {
        printf("Sorry, you failed...\n");
    }

    return 0;
}
```

Switch statement

- The switch statement is a control statement that allows you to test the value of a variable or expression against a list of values (called cases).
- It is often used as an alternative to multiple if-else conditions when checking a variable for equality with several constant values.

C Example_5.c > ...

```c
#include <stdio.h>

int main() {
    char grade = ' ';

    printf("Enter a grade: \n");
    scanf("%c", &grade);

    // Now use the switch statement
    switch(grade)
    {
        case 'A':
        {
            printf("Highest grade\n ");
            break;
        }
        case 'B':
        {
            printf("2nd Highest grade\n ");
            break;
        }
        case 'C':
        {
            printf("3rd Highest grade\n ");
            break;
        }
        default: {
            printf("Sorry, you failed...");
            break;
        } // end default
    }

    return 0;
}
```

Programming Pitfall

1. There is no semi-colon at the end of an if statement
2. Always place a { and } brace around the if, if ..else, case block code when there is even just 1 line of code / 1 statement of code
3. Be careful you do not try to do too much inside an if / if .. else statement

Questions