

Session 3

Variables

Variable types

- There are several different types of variables, and each type of variable is used for storing a particular kind of data
 - integer numerical values
 - noninteger numerical values (e.g 12.032)
 - character values (e.g 'a', "ala bala portocala")

Integer variables

- an integer is any whole number without a decimal point
 - 123, 10999999, 200091, 88, 1
- example of numbers which are **not** integers:
 - 1.234, 999.9, 2.0, -0.034, 3.141523

```
#include <stdio.h>
int main(void)
{
    int salary; // Declare a variable called salary
    salary = 10000; // Store 10000 in salary
    printf("My salary is %d.\n", salary);
    return 0;
}
```

- The = operator is called the assignment operator because it assigns the value on the right to the variable on the left

```
printf("My salary is %d.\n", salary);
```

- The entire text (also called string) between "" is called a control string because it controls how we want the **salary** variable to be displayed.
- %d is called a **conversion specifier** for the value of the variable
 - A **conversion specifier** determines how a binary value should be converted before it is displayed.
 - d stands for decimal (base 10) number

```
#include <stdio.h>
#include <stdlib.h>
```

```
int main()
{
    int brides;
    int brothers;

    brides = 4;
    brothers = 8;

    printf("%d brides for %d brothers\n", brides, brothers);
    return 0;
}
```

- Example of code computing number of citizens:

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int total_citizens;
    int us_citizens;
    int romanians;
    int british;
    int french;

    us_citizens = 10;
    romanians = 23;
    british = 2;
    french = 5;

    // Compute the total number of citizens
    total_citizens = us_citizens + romanians + british + french;
    printf("We have %d citizens in this building\n", total_citizens);
    return 0;
}
```

- Note: we can also assign values to a variable when we declare it (e.g `int age = 29`).
 - This is also a good practice because until we give a value to a variable, it will contain junk values.

Basic arithmetic operations

- An arithmetic expression is any expression that results in a numerical value.
- Example of arithmetic expressions:
 - $1 + 2$
 - 5
 - `total_citizens`

- romanians + french
 - -4
- Most common arithmetic operators

Operator	Action
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus (or remainder operator)

Examples using common arithmetic operators

1. Subtraction and Multiplication - Calories calculator

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int cookies = 5;
    int cookie_calories = 125;
    int total_eaten = 0;

    int eaten = 2;
    cookies = cookies - eaten;
    total_eaten = total_eaten + eaten;
    printf("\nI have eaten %d cookies. There are %d cookies left", eaten,
cookies);

    eaten = 3;
    cookies = cookies - eaten;
    total_eaten = total_eaten + eaten;
    printf("\nI have eaten %d more. Now there are %d cookies left\n", eaten,
cookies);
    printf("\nTotal energy consumed is %d calories.\n", total_eaten *
cookie_calories);

    return 0;
}
```

2. Division and the Modulus Operator

- Statement: Suppose you have a jar of 45 cookies and a group of seven children. You'll divide the cookies equally among the children and work out how many each child has. Then you'll work

out how many cookies are left over

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int cookies = 45;
    int children = 7;
    int cookies_per_child = 0;
    int cookies_left_over = 0;

    //Calculate how many cookies each child gets when they are divided
up    cookies_per_child = cookies / children;
    printf("You have %d children and %d cookies\n", children, cookies);
    printf("Give each child %d cookies.\n", cookies_per_child);

    // Calculate how many cookies are left over
    cookies_left_over = cookies % children;
    printf("There are %d cookies left over.\n", cookies_left_over);

    return 0;
}
```

Exercises:

1. What is a variable?
2. What kind of data can we store inside a variable?
3. Write a C function which returns the addition of two numbers received as parameters.
4. Write a C function which returns the subtraction of two numbers received as parameters.
5. Write a C function which returns the multiplication of two numbers received as parameters.
6. Write a C function which returns the division of two numbers received as parameters.
7. Write a C function which returns the modulus of two numbers received as parameters.
8. Write a C program to find the third angle of a triangle if two angles are given.