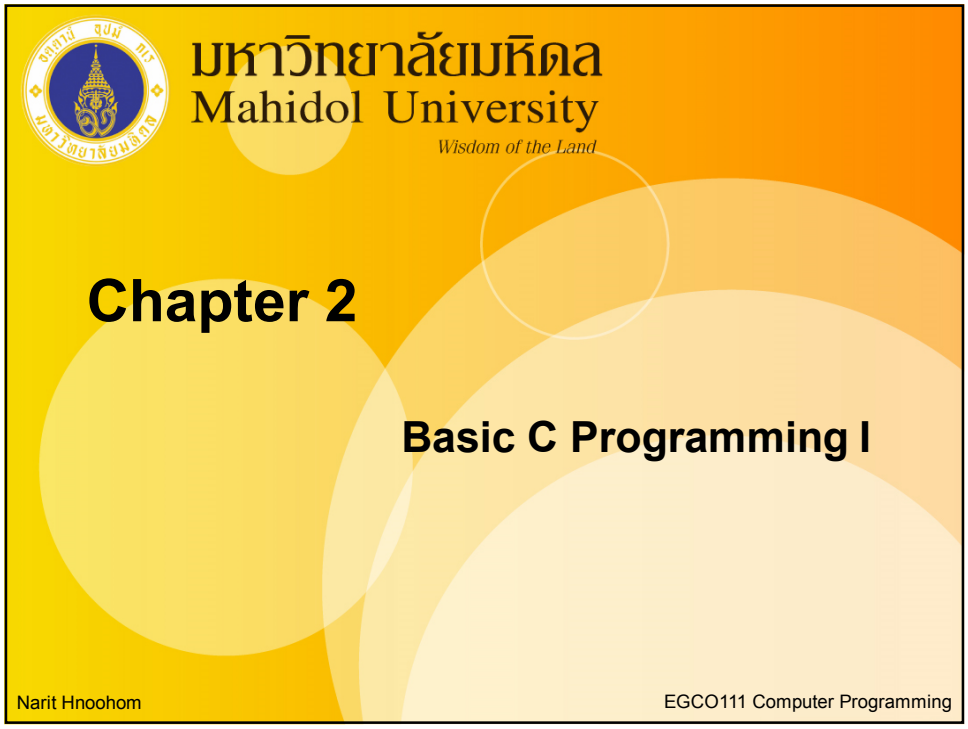

Structure of C Programming

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float  miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```

- Comments
- Preprocessor Directive
- Character Sets
- Data types
- Variables
- Standard Identifiers
- Format Output & Flags
- Escape Sequence

Comments

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```


Comments

- Single Line
- Multiple Line

```
#include <stdio.h>

main()
{
        // Comment only one line
        /* Comment
                many
                        line */
}
```

Single Line

Multiple Line

Preprocessor Directive

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers\n.", kms);
}
```


Preprocessor Directive

- **Directive Include**
  #include <standard_lib> (Read files from directory defined)
  #include "header_file" (Read files from current directory or defined)
- **Example**
  #include <stdio.h>
  #include "sample.h"
- **Directive Define**
  #define *constant_name* *value*
- **Example**
  #define KMS_PER_MILE 1.609
  #define WEEKDAY 5

Constants

Constants are the terms that cannot be changed during the execution of a program.

- Preprocessor Directive
  Example: #define KMS_PER_MILE 1.609
- Reserve word: const
  Example: const float PI = 3.14159;

```
#include <stdio.h>
#define KMS_PER_MILE      1.609
```

Preprocessor Directive

```
main()
{
      const float PI = 3.14159;
}
```

Reserve Word

Main Function

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```

Main Function

- Main function or Initial function of C programming
- All programs must have this function.

```
main()
{
        /* Your Codes */
}
```


Character Sets

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float  miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```

Character Sets

- Letter (52)
  - A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
  - a b c d e f g h I j k l m n o p q r s t u v w x y z
- Decimal Digits (10)
  - 0 1 2 3 4 5 6 7 8 9
- Punctuation Marks (29)
  - ! " # % & ' ( ) * + , - . / : ; < = > ? [\ ] ^ _ { | } ~
- White Space Characters (5)
  - space, horizontal tab, vertical tab, new line, and form feed
- Separators (6)
  - Parentheses ( )
  - Braces { }
  - Bracket [ ]
  - Semicolon ;
  - Comma ,
  - Full Stop .

Data Types

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```

Data types

Type	Storage Size (byte)	Min Value	Max Value
char	1	0	255
unsigned char	1	0	255
signed char	1	-128	127
int	2	-32,768	32,767
unsigned int	2	0	65,535
short	2	-32,768	32,767
unsigned short	2	0	65,535
long	4	-2,147,483,648	2,147,483,647
unsigned long	4	0	4,294,967,295

Type	Storage Size (byte)	Value Range	Precision
float	4	±3.4E+38	6
double	8	±1.7E+308	15
long double	10	±1.1E+4932	19

Identifier

- **User-defined Identifier**
  - Function's Name
  - Variable's Name
- **Standard Identifier**
  - Come with C compiler
  - Is in C library
  - May be difference (depend on the company that makes each complier)

Function Name

```
#include <stdio.h>
Miles2Kilometers(float a);
main() {
        float miles;
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        Miles2Kilometers(miles);
}
Miles2Kilometers(float a) {
      float kms;
      const float KMS_PER_MILE = 1.609;
      kms = KMS_PER_MILE * a;
      printf("That equals %f kilometers.\n", kms);
}
```


Variable Name

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```

Variable Name

- **Case Sensitive**
  - TEST, test, Test, tEsT, TEst
- **First Letter : character or underscore "_"**
  - Other letters : character, number, or underscore
- **No Space**
- **No Special Characters**
  - Such as … $ @ # &
- **Not be Reserved Words**

Reserve Word

_cs	_ds	_es	_ss
asm, auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

Exercise of Variable Name

- **These variable names are correct or incorrect ?**
  1. X
  2. x
  3. x2
  4. 2xyz
  5. 44aa
  6. _myclass
  7. G2000
  8. 3000
  9. helLo_2worlD
  10. default

Variable Declaration

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```

Variable Declaration

- **Syntax**
  variable_type variable name,[variable_name];

- **Example (declaration)**
  char c;
  int a;
  float x,y,z;

- **Example (declaration and Initialization)**
  char c = 'a';
  int a = 0;
  float x,y,z = 2.35;

Prohibitions

- **Rules for setting a variable name (Prohibitions)**
  - It is not reserved words or standard identifier.
  - It is not begin with number, such as
    int 1X;
  - It has not white space, such as
    int X 1;
  - It does not use any punctuation
    (except_), such as
    int 1+x;

Standard Identifier

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```


Standard Identifier

- **Standard Identifier**
  scanf(), printf()
- **Standard Library**
  #include <stdio.h>

- **Syntax**
  printf (const char *format [,argument,…]);

- **Example**
  printf("Sawasdee EGCO111");
  printf("Mahidol University");
  printf("2 + 3 = 5");

- **Result**
  Sawasdee EGCO111
  Mahidol University
  2 + 3 = 5

Escape Sequence

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```


Escape Sequence

Escape Sequence	Description	ASCII
\t	(Horizontal) Tab	011
\v	Vertical Tab	
\n	New Line	012
\r	Return	015
\f	Form Feed	014
\b	Backspace	010
\\	Back Slash	092
\'	Single Quote	060
\"	Double Quote	148
\0	Null	0

Example of printf()

- **Example**

```
1  #include <stdio.h>
2  main()
3  {
4    printf("Mahidol");
5    printf("\n\tUniversity");
6    printf("\n");
7    printf("\\\"");
8  }
```

- **Result:**

```
Mahidol
        University
\"
```


Exercise of printf()

- **Write a program to display the following text.**
- **Example of result:**

```
1st SEMESTER 2014    \566258  Mr. cee rukrean

'SCPY 151' General Physics I          3   "A"
'EGCO 111'  Computer programming      3   "A"
'SCMA 115' Calculus                   3   "B"
-----------------------------------------------------------------
                        cum : 3.66
-----------------------------------------------------------------
```

Variable Displaying

- **Syntax**

```
printf( "a=%d and b=%d" , a, a/2);
```

Format String — Arguments

- **Example**

```
int a = 10;
printf("a=%d and b=%d", a, a/2);
```


Variable Displaying

- **Syntax**

```
printf( "a=%d and b=%d" , a, a/2);
```

Format String — Arguments

- **Example**

```
int a = 10;
printf("a=%d and b=%d", a, a/2);
```

- **Result**

a=**10** and b=**5**

Example of printf: Variable

- **Example**

```
1  #include <stdio.h>
2  main()
3  {
4     int x;
5     x=345;
6     printf("x = %d",x);
7  }
```

- **Result:**
  x = **345**

Example of printf: Variable

- **Example**

```
1  #include <stdio.h>
2  main()
3  {
4    int x;
5    int y;
6    int sum;
7    x = 3;
8    y = 5;
9    sum = x + y;
10 printf(“sum = %d", sum);
   }
```

- **Result:**
  sum = 8

Example of printf: Variable Declaration

- **Example**

```
1  #include <stdio.h>
2  main()
3  {
4    int x = 3;
5    int y = 5;
6    int sum = x + y;
7    printf("sum = %d", sum);
8  }
```

- **Result:**
  sum = 8

Variable VS. Variable Declaration

```
#include <stdio.h>
main()
{
  int x;
  int y;
  int sum;
  x = 3;
  y = 5;
  sum = x + y;
printf("sum = %d", sum);
}
```

```
#include <stdio.h>
main()
{
  int x = 3;
  int y = 5;
  int sum = x + y;
  printf("sum = %d", sum);
}
```

- **Result:**
  sum = 8

Format Output & Flags

```
// Converts distances from miles to kilometers.
#include <stdio.h>
#define KMS_PER_MILE 1.609
main() {
        float miles, kms;
        /* Get the distance in miles. */
        printf("Enter the distance in miles : ");
        scanf("%f", &miles);
        /* Convert the distance to kilometers. */
        kms = KMS_PER_MILE * miles;
        /* Display the distance in kilometers. */
        printf("That equals %f kilometers.\n", kms);
}
```


Format Output & Flags: String

Specifier	Argument Types	Representation
%c	char	display a character
%s	string	display sentence or word

Example

```
int score = 120;
char player[]="Mary";
printf("%s has %d points.\n", player, score);
```

Result

Mary has **120** points.

Exercise of printf(): String

- **Exercise1**

```
1  #include <stdio.h>
2  main()
3  {
4     char X = 'x';
5     printf("Why is %cyz", X);
6  }
```


Exercise of printf(): String

- **Exercise2**

```
1  #include <stdio.h>
2  main()
3  {
4     int a = 3;
5     char b[5] = "nine";
6     printf("%d x %d= %s", 3, a, b);
7  }
```

Exercise of printf(): String

- **Exercise3**

```
1  #include <stdio.h>
2  main()
3  {
4     char A[2] = 'a';
5     printf("Have %s nice day", A);
6  }
```


Format Output & Flags: Integer

Specifier	Argument Types	Representation
%d, %i	int	display integer base 10
%u	unsigned int	display integer base 10
%o	unsigned int	display integer base 8
%x	unsigned int	display integer base 16 (lowercase)
%X	unsigned int	display integer base 16 (uppercase)

Example

```
printf(" %4d %4o %4x %4X\n", 63, 63, 63, 63);
```

Result

```
  63   77   3f   3F
```

Exercise of printf(): Integer

- **Exercise**

```
1  #include <stdio.h>
2  main()
3  {
4    int x = 6;
5    int y = 8;
6    int multiple = x * y;
7    printf("%d * %d = %d", y, x, multiple);
8  }
```


Format Output & Flags: Floating-point

Specifier	Argument Types	Representation
%f	double	display floating-point base 10
%e, %E	double	exponential form
%g, %G	double	floating-point or exponential form (display the shorter form)
%a, %A	double	exponential form base 16

Example

```
double x = 12.34;
printf(" %f %e %E %.1f\n", x, x, x, x);
```

Result

```
12.340000  1.234000e+01  1.234000E+01  12.3
```

Example of printf() : Floating-point

- **Example**

```
1  #include <stdio.h>
2  main()
3  {
4     printf("\n%10s %5d %5.2f", "Sompong", 8, 29.13);
5     printf("\n%10s %5d %5.2f", "Somsak", 150, 2.5);
6     printf("\n%10s %5d %5.2f", "Somboon", 29, 155.158);
7  }
```

- **Result:**

```
   Sompong     8   29.13
    Somsak   150    2.50
   Somboon    29  155.16
```


Exercise of printf(): Floating-point

- **Exercise1**

```
1  #include <stdio.h>
2  main()
3  {
4     printf("%d\n", 111);
5     printf("%06d\n", 111);
6     printf("%f\n", 111.22);
7     printf("%.2f\n", 111.22);
8  }
```

Exercise of printf(): Floating-point

- **Exercise2**

```
1  #include <stdio.h>
2  main()
3  {
4     printf("1234567890\n");
5     printf("%-10.2f\n", 12345.90);
6     printf("%10.2f\n", 12345.90);
7  }
```


Exercise of printf(): Floating-point

- **Exercise3**

```
1  #include <stdio.h>
2  main()
3  {
4     int age = 18;
5     float weight = 57.6;
6     float height;
7     height = 176.3;
8     printf("Mr.cee rukrean is %d years old",age);
9     printf(", weight %0.1f kilograms and tall %0.1f cms.\n",weight,height);
10 }
```

Exercise of printf()

- **Write a program to convert temperature from Fahrenheit to Celsius.**
  - Formula: Celsius = (Fahrenheit-32) x 5/9
- **Example of result:**
  98.6 Fahrenheit **= 37** Celsius

Exercise of printf()

- **Write a program to convert a number of days into a number of months and days**
  - **Assume 1 month = 30 days**
- **Example of result:**
  1825 days = **60** months and **25** days

Exercise of printf()

- **Write a program to swap the value of two numbers (A and B)**
- **Example of result:**
  Before: a = **3,** b = **4**
  After: a = **4,** b = **3**

Exercise

- **Write a program to display the following text.**
- **Example of result:**

```
Radius of circle is 2.500000
            Area of circle is 15.707
                    Circumference of circle is 19.63493
Conclusion
************************************************
Radius              Area              Circular
************************************************
2.50                15.70             19.63
************************************************
************************************************
1234567890      1234567890      1234567890
```

Thanks for your attention