

Introduction to Programming

Lecture Two

N.R.Aravind

I.I.T. Hyderabad

19 Sep 2017

Lab guidelines

- 1 Write programs on paper before the lab.
- 2 Save programs (pen-drive, email, g-drive)
- 3 Meaningful program names:
addition.c, circle.c, helloWorld.c, quiz.c,
maximum.c
- 4 Avoid: abcd.c, ramakrishna.c,
cs15btech1001.c, program1.c, program2.c
- 5 Meaningful variable names:
radius, area, count, username, country, length
- 6 Take breaks!

Linux commands

- ls, cd, clear
- man
-

Topics in this lecture

- **Review**
- `if (--) {...} else {...}`
- `while (--) {...}`

Review

Assignment with int

```
#include<stdio.h>
main()
{
    int a,b,c,d;
    a=25;
    b=10;
    c=2*a/b;

    // c=5.

    d=a/b*2;
```

Assignment with int

```
#include<stdio.h>
main()
{
    int a,b,c,d;
    a=25;
    b=10;
    c=2*a/b;

    // c=5.

    d=a/b*2;    //d=4.

    c=a+b/2;
```

Assignment with int

```
#include<stdio.h>
main()
{
    int a,b,c,d;
    a=25;
    b=10;
    c=2*a/b;

    // c=5.

    d=a/b*2;    //d=4.
    c=a+b/2;    //d=30.
}
```


float

- `float num1=2.16789;`
- `printf("%f",num1);`
- `printf("%8.2f ", num1);`

float

- `float num1=2.16789;`
- `printf("%f",num1);`
- `printf("%8.2f ", num1);` `// Prints 2.17`

float

- `float num1=2.16789;`
- `printf("%f",num1);`
- `printf("%8.2f ", num1);` `// Prints 2.17`
- `printf("%8.4f", num1);`

float

- `float num1=2.16789;`
- `printf("%f",num1);`
- `printf("%8.2f ", num1);` `// Prints 2.17`
- `printf("%8.4f", num1);` `//Prints 2.1679`

long and double

- `long num1;`
- `num1 = 26435618;`
- `%ld`

- `double num1;`
- `num1 = 23438.1269;`
- `%f`

Exercise 2: circle.c

```
#include<stdio.h>
#define PI 3.1416
main()
{
    float radius, area, circumference;
    // Input statements
    area=PI*radius*radius;
    circumference=2*PI*radius;
    // Output statements
}
```

char

- `char userName[10];`
- `userName[0]='Z';`
- `char ch;`
- `ch='A'; ch='#'; ch='5';`

char

- `char userName[10];`
- `userName[0]='Z';`
- `char ch;`
- `ch='A'; ch='#'; ch='5';`
- `ch=getchar(); // Input`

char

- `char userName[10];`
- `userName[0]='Z';`
- `char ch;`
- `ch='A'; ch='#'; ch='5';`
- `ch=getchar();` // Input
- `printf("%c",ch);` // Output

char

- `char userName[10];`
- `userName[0]='Z';`
- `char ch;`
- `ch='A'; ch='#'; ch='5';`
- `ch=getchar();` // Input
- `printf("%c",ch);` // Output
- `char` uses 1 byte of memory.

char arrays

```
char text[10]="PROGRAM";
```

Variable	Value
text[0]	P
text[1]	R
text[2]	O
text[3]	G
text[4]	R
text[5]	A
text[6]	M
text[7]	Ø
text[8]	
text[9]	

The modulus operator

```
int a,b,c;  
a=75;  
b=4;  
c=a%b;  
// c=3.  
% is the modulus operator.
```

What day of the week is it 100 days from now?

if (__) { ... } else { ... }

if (_) { ... }

```
int a=10,b=5,c=15;  
if (a > 7)  
{  
    printf(" Hello");  
}  
if (b > 7)  
{  
    printf(" Welcome");  
}  
if (c==15)  
{  
    printf(" Bye");  
}
```

The if (--) {...} statement

Syntax:

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

// If the expression is true, execute the statements inside the block.

if (_ _) { ... }

```
// Accept num from user.  
if ( num < 0 )  
{  
    num=-num;  
}  
printf(" %d",num);
```

if (_ _) { ... } else { ... }

```
int a=5,b=15;
if (a > 7)
{
    printf(" Hello");
}
else
{
    printf(" Welcome");
}
if (b==15)
{
    printf(" Bye");
}
```

The if ... else statement

Syntax:

```
if (expression)
{
    Statements S1
}
else
{
    Statements S2
}
```

// If the expression is true, S1 will be executed,
otherwise S2 will be executed.

The if...else statement

```
char answer;  
int score;  
printf("Who is the Prime Minister of Japan?");  
printf(" a. Naoto Kan");  
printf(" b. Shinzo Abe");  
printf(" c. Akira Kurosawa");  
printf(" d. Hikaru Nakamura");  
printf(" Enter your choice: ");  
answer=getchar();
```

Nested if ... else statement

```
int num1,num2;  
if (num1 > num2)  
    { printf(" The first number is larger." ); }
```

Nested if ... else statement

```
int num1,num2;  
if (num1 > num2)  
    { printf(" The first number is larger."); }  
else {  
    if (num2 > num1)  
        { printf(" The second number is larger."); }  
}
```

Nested if ... else statement

```
int num1,num2;  
if (num1 > num2)  
    { printf(" The first number is larger."); }  
else {  
    if (num2 > num1)  
        { printf(" The second number is larger."); }  
    else  
        { printf(" The numbers are equal.") }  
}
```

The whileif(--) {...} statement

Syntax:

```
while (expression)
{
    statements
}
```

// While the expression is true, execute the statements inside the block.

//Until the expression is false, execute the statements inside the block.

Lab guidelines: Reminder!

- 1 Write programs on paper before the lab.
- 2 Save programs (pen-drive, email, g-drive)
- 3 Meaningful program names:
addition.c, circle.c, helloWorld.c, quiz.c,
maximum.c
- 4 Avoid: abcd.c, ramakrishna.c,
cs15btech1001.c, program1.c, program2.c
- 5 Meaningful variable names:
radius, area, count, length, username, country
- 6 Take breaks!

That's all, folks!