Introduction to Programming

Basics of Programming (3) : Branches

Waseda University

Today's topics

- How to use Branches
  - if else statement
  - switch statement
- How to use logical operators
  - AND "&&"
  - OR "||"
  - NOT " ! "

Exercise 1

bmi1.c: The program for calculating BMI

BMI is a value calculated from the relationship between your height and weight. BMI is calculated as "10^4" x "body weight[kg]" x height[cm]^2.

Input your height[cm]: 173.2 【*Enter*】
Input your weight[kg]: 60.3 【*Enter*】
Your BMI is 20.1.

- $10^4 = 10000.0$
- Don't forget to change the directory 【cd work】
- For editing 【emacs bmi1.c &】
- For compiling 【gcc bmi1.c】
- For running 【./a.out】

Hints for Exercise 1

- Declaring variables:

  ```
  double Height, Weight, BMI;
  ```

- Display: `printf("  ")`
- Input: `scanf("  ")`

  ```
  scanf("%lf",&Height);
  ```

- Calculation of BMI：

  ```
  BMI = 10000.0*Weight / (Height*Height);
  ```

- Output the BMI :

  ```
  printf("%f", BMI);
  ```

if statement (Branches)

How to use if statement

```
if (condition){
 If the condition is true, this statement is executed.
}else{
 If the condition is false, this statement is executed.
}
```

- If you have one thing you want it to execute, you can omit {}.
- If there's nothing you want it to do after "else" you can omit them.
- You can also nest an "if" statement within an "if" statement and establish a complex conditional.
- By ordering your indents, you'll make it easier to know where the statements belong in the structure.

if statement

Example:

```
if (x>0) printf("x is a positive number ¥n");
if (x>0) {
    printf("x is a positive number ¥n");
}
```

if statement

Relational operators "==, !=, <, >, <=, >="

- Conditional expressions describe the equation that will be a conditional using a relational operator.
- It takes the variable if true "1", if false "0".

a==b	If a is equal to b, it holds "1"
a!=b	If a is not equal to b, it holds "1"
a<b	If a is less than b, it holds "1"
a>b	If a is greater than b, it holds "1"
a<=b	If a is less than or equal to b, it holds "1"
a>=b	If a is greater than or equal to b, it holds "1"

- x is equal to "10": if (x==10) ...
  Note: "=" means Assignment, "==" is equal.
- x is greater than or equal to 20: if (x>=20) ...
- Conditional expressions can also use arithmetic operators like "a*b+c< a/b"

if statement

Logical operators " | |, &&, !"

- Conditional expressions can use logical operators to describe several conditions.

a || b	It returns "1" if any one of its two arguments are true
a && b	It returns "1" if its two arguments are both true
! a	It returns "1" if a is true, It returns "0" if a is true

- x is greater than or equal to 10 and less than 20 if (x>=10 && x<20) ...

Logical operations

- A logical operation is a operation for two kinds of variables: true and false.
- `x>=3 && x<5`:
  if x is greater than or equal to 3 and x is less than 5, it holds true
- `x>=3 || x<5`:
  if x is greater than or equal to 3 or x is less than 5, it holds true
  Note: In technical jargon it expresses as true when both sides are true. Meaning it comprises of "both" conditions.
- 「`!(x==3)`」: If x is not 3, it holds true

Example

- "`!(x==0 || y==0)`", "`x!=0 && y!=0`" This has the same meaning

  Note: You might use "parentheses ( )" to make it easier to read.

  If you don't put in "parentheses ()", the processing order is the followings: "`!`", "`<, >, <=, >=`", "`==, !=`", "`&&`", "`!!`"

Example of if statement 1

```
#include <stdio.h>
int main(void){
    double Score = 85;

    if(             Score<80) printf("Your grade is B¥n");
    if(80<=Score && Score<90) printf("Your grade is A¥n");
    if(90<=Score            ) printf("Your grade is A++¥n");

    return 0;
}
```

- "If statements" are decided with the following table:

Score<80	Your grade is B
80<=Score and Score<90	Your grade is A
90<=Score	Your grade is A++

- In the 2nd if statement, if the left and right side conditions are true then it becomes true.

Example of if statement 2

```
#include <stdio.h>
int main(void){
    double Score = 85;
    if(Score < 80)
        printf("Your grade is B¥n");
    else if(Score < 90)
        printf("Your grade is A¥n");
    else
        printf("Your grade is A++¥n");
    return 0;
}
```

- This example outputs the same result of "Example of if statement 1"
- "If else" statement nested after the first "else"

Score $<$ 80	Yes			Your grade is B
	No	Score $<$ 90	Yes	Your grade is A
			No	Your grade is A++

Example of if statement 3

```
if(Score< 70){
    if(Score< 60) printf("Your grade is D¥n");
    else          printf("Your grade is C¥n");
}else{
    if(Score< 80) printf("Your grade is B¥n");
    else{
        if(Score< 90) printf("Your grade is A¥n");
        else          printf("Your grade is A++¥n");
    }
}
```

- This is a more complicated conditional branch but you can process them all at once

<table>
<tr><td rowspan="8">Score<70</td><td rowspan="2">Yes</td><td rowspan="2">Score<60</td><td>Yes</td><td colspan="3">Your grade is D</td></tr>
<tr><td>No</td><td colspan="3">Your grade is C</td></tr>
<tr><td rowspan="6">No</td><td rowspan="6">Score<80</td><td>Yes</td><td colspan="3">Your grade is B</td></tr>
<tr><td rowspan="5">No</td><td rowspan="5">Score< 90</td><td>Yes</td><td>Your grade is A</td></tr>
<tr><td>No</td><td>Your grade is A++</td></tr>
</table>

Exercise 2

bmi2.c: The program to displays WHO standard BMI scores

Create a program that displays WHO standard BMI scores using "if" statements.

BMI	WHO standard
18.5<BMI	Underweight
18.5<=BMI<25.0	Normal
25.0<=BMI<30.0	Pre-obese
30.0<=BMI	Obese class

Input your height[cm]: 173.2 【Enter】
Input your weight[kg]: 60.3 【Enter】
Your BMI is 20.1. Normal.

switch statement

How to use switch statement

```
switch (expression){
    case constant 1:
        statement;
        break;
    case constant 2:
        statement;
        break;
    case constant 3:
        statement;
        break;
    default:
        statement;
        break;
}
```

- You can divide the processes you want it to perform by the variables of an integer equation in parentheses following "switch" statements. (Be careful
- because it can't handle real numbers)
- If there's not a match then it will execute default statement.
- It's possible to omit the default statement.
- If you don't have "break;", it will continue executing the "default" statement so don't forget to include "break;"

Example by using switch statement

```
#include <stdio.h>
int main(void){
    int num;
    scanf("%d", &num);
    printf("%d is divisible by 3 with",num);

    switch (num % 3){
        case 1 : printf("remainder 1¥n"); break;
        case 2 : printf("remainder 2¥n"); break;
        default: printf("remainder 0¥n"); break;
    }

    return 0;
}
```

- "num % 3" is a remainder of "num divide 3".
- Depending on a value of "num % 3", it will display the different character string

Summary

- How to use Branches
  - if else statement
  - switch statement
- How to use logical operators
  - AND "&&"
  - OR "||"
  - NOT " ! "