



ບົດທີ 5

ຄໍາສັ່ງຄວບຄຸມ Control Statement





ເນື້ອໃນຫຍໍ້

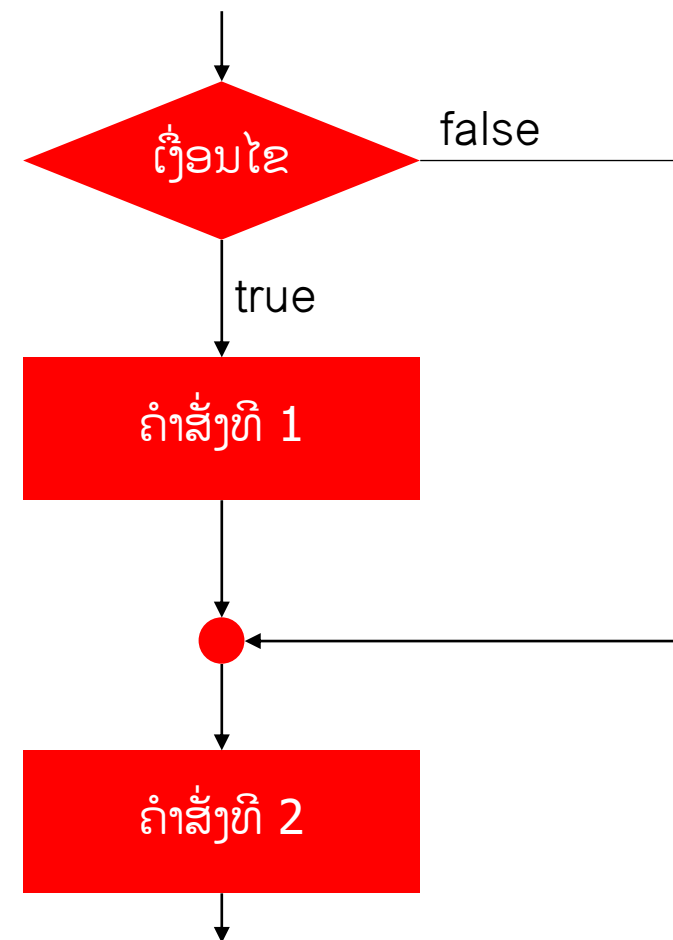
- I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement).
- II. ຄໍາສັ່ງທຳຊ້າ (Iteration Statement).



I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement)

ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ເປັນຄໍາສັ່ງທີ່ໃຊ້ເລືອກຫຼັກໂດຍພິຈາລະນາຈາກເງື່ອນໄຂທີ່ກຳນົດ. ຄໍາສັ່ງ **if** ເງື່ອນໄຂດຽວ.

```
If (ເງື່ອນໄຂ){  
    ຄໍາສັ່ງທີ 1;  
}  
ຄໍາສັ່ງທີ 2;
```





I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່

ຄໍາສັ່ງ if ເງື່ອນໄຂດຽວ ຕົວຢ່າງການຂຽນ code

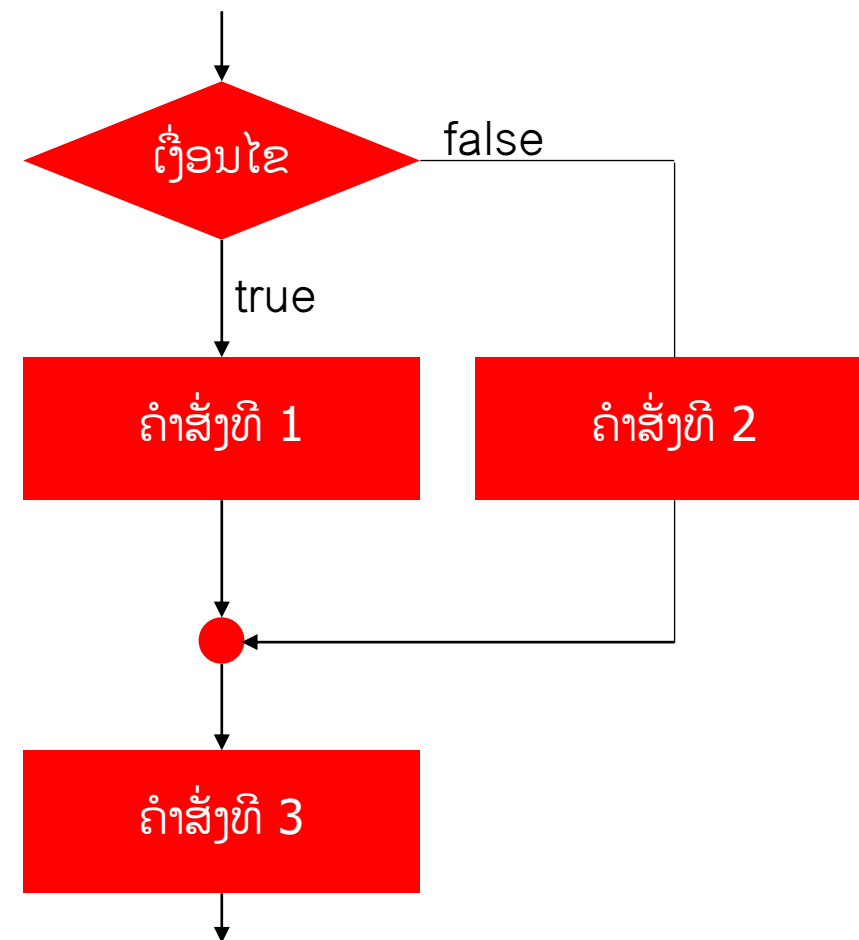
```
[*] example1.c example11.c
1  #include<stdio.h>
2  #include<stdlib.h>
3
4  main(){
5      printf("\t@@@@@@@@ lesson 5 example 1 @@@@@@@@@\n");
6      system("color f1");
7
8      int age;
9      printf("\tHow old are you ?\n");
10     printf("\tEnter your age:");
11     scanf("%d",&age);
12     if(age<18){
13         printf("\tyou are young\n");
14     }
15     printf("\tyou are %d yeas old",age);
16
17     printf("\t@@@@@@@@ lesson 5 example 1 @@@@@@@@@\n");
18 }
```



I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່

ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ເປັນຄໍາສັ່ງທີ່ໃຊ້ເລືອກຫຳໂດຍພິຈາລະນາຈາກເງື່ອນໄຂທີ່ກຳນົດ. ຄໍາສັ່ງ **if else** ເງື່ອນໄຂດຽວ.

```
If (ເງື່ອນໄຂ){  
    ຄໍາສັ່ງທີ 1;  
}  
else{  
    ຄໍາສັ່ງທີ 2;  
}  
ຄໍາສັ່ງທີ 3;
```





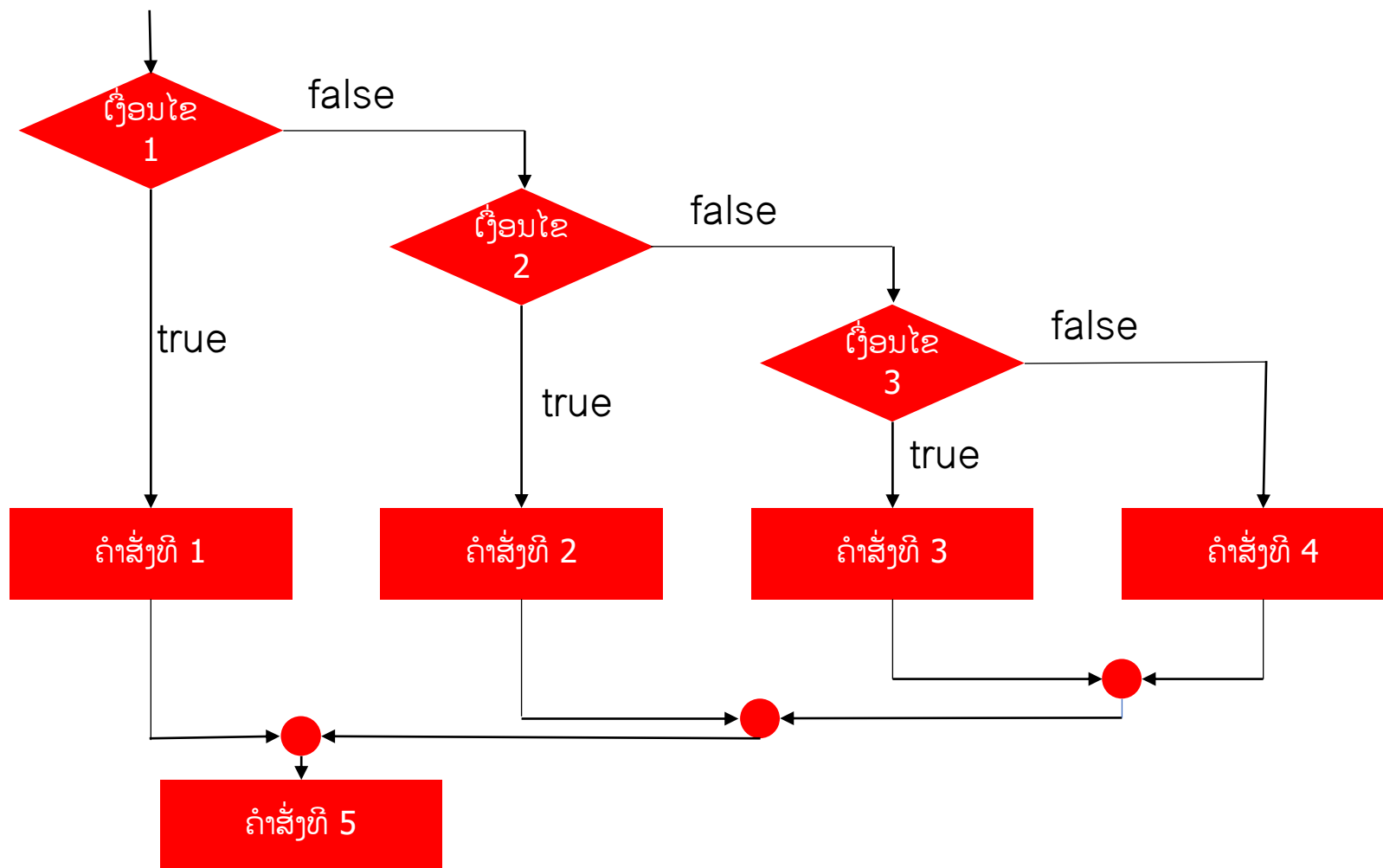
I. ຄຳສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່

ຄຳສັ່ງເງື່ອນໄຂ (Condition Statement) ເປັນຄຳສັ່ງທີ່ໃຊ້ເລືອກທຳໂດຍພິຈາລະນາຈາກເງື່ອນໄຂທີ່ກຳນົດ.
ຄຳສັ່ງ if ຊອນ if ຫຼາຍເງື່ອນໄຂ.

```
if (ເງື່ອນໄຂ 1){  
    ຄຳສັ່ງທີ 1;  
}  
else if (ເງື່ອນໄຂ 2){  
    ຄຳສັ່ງທີ 2;  
}  
else if (ເງື່ອນໄຂ 3){  
    ຄຳສັ່ງທີ 3;  
}  
Else{  
    ຄຳສັ່ງທີ 4;  
}  
    ຄຳສັ່ງທີ 5;
```



I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່ ຄໍາສັ່ງ if ຊອນ if ຫຼາຍເງື່ອນໄຂ.





I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່ ຄໍາສັ່ງ if ຊອມ if ຫຼາຍເງື່ອນໄຂ.

```
[*] example1.c  example11.c  example2.cpp  activity2.c  example3.c
1  #include<stdio.h>
2  #include<stdlib.h>
3  main(){
4      printf("\t@@@@@@@@@@ lesson 5 example 3 @@@@@@@@@@@\n\n");
5      system("color f1");
6      int points;
7      printf("\tPlease Enter Points:");
8      scanf("%d",&points);
9      if(points>=80){
10         printf("\tCongratulations\n");
11         printf("\tC language programming subject\n");
12         printf("\tYou get grade A\n");
13     }
14     else if(points>=70){
15         printf("\tYou get grade B\n");
16     }
17     else if(points>=60){
18         printf("\tYou get grade C\n");
19     }
20     else if(points>=50){
21         printf("\tYou get grade D\n");
22     }
23     else{
24         printf("\tYou get grade F\n");
25     }
26     printf("\tSee you again! next course\n\n");
27     printf("\t@@@@@@@@@@ lesson 5 example 3 @@@@@@@@@@@\n\n");
28 }
```




I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່

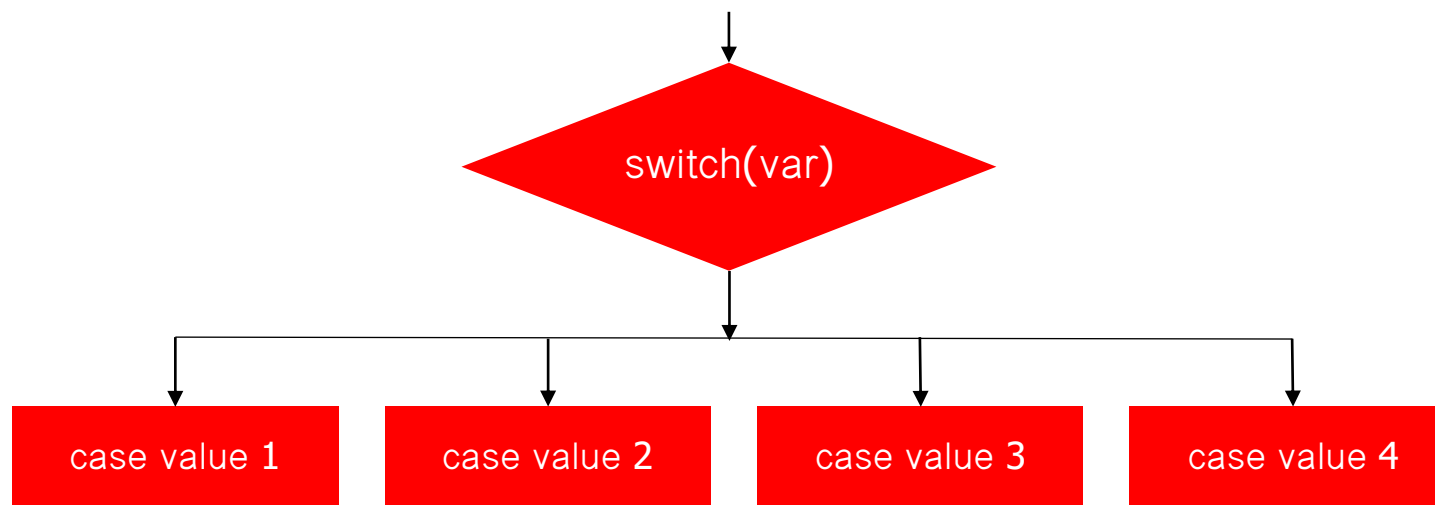
ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ເປັນຄໍາສັ່ງທີ່ໃຊ້ເລືອກທໍາໂດຍພິຈາລະນາຈາກເງື່ອນໄຂທີ່ກຳນົດ.
ຄໍາສັ່ງ if ຊອນ if ຫຼາຍເງື່ອນໄຂ.

ຄະແນນ	ເກຣດ
ຄະແນນ ≥ 80	A
ຄະແນນ ≥ 75	B+
ຄະແນນ ≥ 70	B
ຄະແນນ ≥ 65	C+
ຄະແນນ ≥ 60	C
ຄະແນນ ≥ 55	D+
ຄະແນນ ≥ 50	D
ຄະແນນ ≥ 0	F



I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່

ຄໍາສັ່ງ switch-case: ເປັນຄໍາສັ່ງທີ່ໃຊ້ເລືອກເຮັດວຽກຕາມຄໍາສັ່ງຕ່າງໆ ໂດຍພິຈາລະນາຈາກຄ່າຂອງຕົວປຽນ ຫຼື ຜົນທີ່ກໍານົດໃຫ້ຕົງກັບ case ໃດ. ຄໍາສັ່ງ switch ຈະກວດສອບຄ່າຂອງຕົວປຽນ ຫຼື ຜົນທີ່ກໍານົດແລ້ວພິຈາລະນາຕົງກັບ case ໃດກໍ່ຈະເຮັດວຽກທີ່ຢູ່ພາຍໃຕ້ຂອງຄໍາສັ່ງຂອງ case ນັ້ນ, ແຕ່ຖາກວດສອບແລ້ວບໍ່ຕົງຕາມຄໍາສັ່ງໃດໆໃນ case ແມ່ນຈະເຂົ້າສູ່ການເຮັດວຽກພາຍໃຕ້ສ່ວນຂອງ default.





I. ຄໍາສັ່ງເງື່ອນໄຂ (Condition Statement) ຕໍ່ ຄໍາສັ່ງ switch-case

```
switch (){  
    case ຄາທີ 1;  
        ຄໍາສັ່ງທີ 1;  
        break;  
    case ຄາທີ 2;  
        ຄໍາສັ່ງທີ 2;  
        break;  
    case ຄາທີ 3;  
        ຄໍາສັ່ງທີ 3;  
        break;  
    default :  
        ຄໍາສັ່ງທີ 4;  
}
```



II. ຄໍາສັ່ງທຳຊ້າ (Iteration Statement).

ເປັນຄໍາສັ່ງທີ່ໃຊ້ກຳນົດວົງຮອບ (loop) ໃນການເຮັດວຽກຂອງໂປຣແກຣມ, ເຊິ່ງມີປະໂຫຍດຫຼາຍໃນກໍລະນີການເຮັດວຽກແບບຊ້າກັນຫຼາຍໆຄັ້ງ.

❑ ຄໍາສັ່ງເຮັດຊ້າ for

```
For( ກຳນົດຄ່າເລີ່ມຕົ້ນໃຫ້ກັບຕົວປຽນ; ເງື່ອນໄຂທີ່ຕ້ອງການກວດສອບ; ປັບຄ່າຂອງຕົວປຽນ) {  
    ຄໍາສັ່ງທີ 1;  
}  
    ຄໍາສັ່ງທີ 2;
```

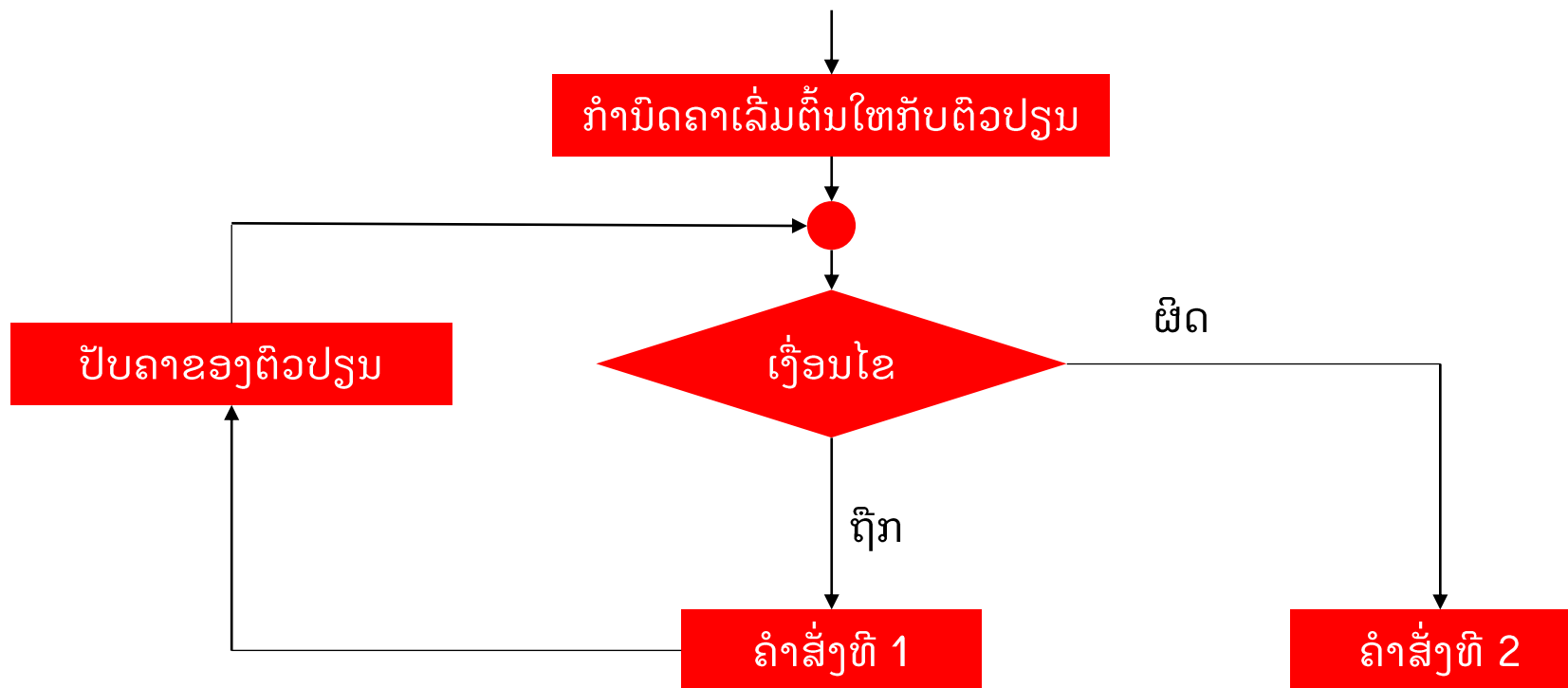
- ກຳນົດຄ່າເລີ່ມຕົ້ນໃຫ້ກັບຕົວປຽນ: ສ່ວນນີ້ແມ່ນການກຳນົດຄ່າເລີ່ມຕົ້ນໃຫ້ກັບຕົວປຽນ ທີ່ຕ້ອງການໃຊ້ງານໃນ loop ໂດຍປົກກະຕິເຮົາຈະໃຊ້ຕົວປຽນນີ້ເປັນຕົວຄວບຄຸມ loop.
- ເງື່ອນໄຂທີ່ຕ້ອງການກວດສອບ: ເປັນເງື່ອນໄຂໃນການເຮັດວຽກຂອງ loop, ຖ້າເງື່ອນໄຂນີ້ເປັນຄ່າຖືກຈະເຮັດວຽກພາຍໃນ loop, ແຕ່ຖ້າເງື່ອນໄຂນີ້ເປັນຄ່າຜິດກໍ່ອອກຈາກ loop ໄປເຮັດວຽກຢູ່ນອກຄໍາສັ່ງ loop.
- ປັບຄ່າຂອງຕົວປຽນ: ເປັນຄໍາສັ່ງທີ່ໃຊ້ປັບຄ່າຂອງຕົວປຽນໃຫ້ກັບ loop ໂດຍອາດເປັນການເພີ່ມຄ່າ ຫຼື ລົດຄ່າຂອງຕົວປຽນນັ້ນ, ເຊິ່ງສ່ວນນີ້ເປັນສ່ວນທີ່ມີຄວາມສຳຄັນຫຼາຍ.



II. ຄຳສັ່ງທຳຊ້າ (Iteration Statement).

ເປັນຄຳສັ່ງທີ່ໃຊ້ກຳນົດວົງຮອບ (loop) ໃນການເຮັດວຽກຂອງໂປຣແກຣມ, ເຊິ່ງມີປະໂຫຍດຫຼາຍໃນກໍລະນີການເຮັດວຽກແບບຊ້າກັນຫຼາຍໆຄັ້ງ.

❑ ຄຳສັ່ງເຮັດຊ້າ for





II. ຄໍາສັ່ງທຳຊ້າ (Iteration Statement).

❑ ຄໍາສັ່ງເຮັດຊ້າ for

```
example16.c  example6.c
1  #include<stdio.h>
2  #include<stdlib.h>
3  main(){
4      printf("\t===== example 6 =====\n");
5      system("color f4");
6      int i;
7      for(i=1;i<=12;i++){
8          printf("\t 2 x %d\t=\t%d\n",i,2*i);
9      }
10     printf("\t===== example 6 =====\n");
11 }
```



II. ຄໍາສັ່ງທຳຊ້າ (Iteration Statement).

- ❑ ຄໍາສັ່ງເຮັດຊ້າ **for** ຊອນ **for**: ແມ່ນການນຳເອົາຄໍາສັ່ງທຳຊ້າຫຼາຍໆຊຸດມາເຮັດວຽກຊອນກັນ, ເຊິ່ງການເຮັດວຽກແບບນີ້ແມ່ນຈະເລີ່ມຈາກ loop for ທີ່ຢູ່ຂາງນອກ (outer loop) ໄປສູ່ການເຮັດເຮັດວຽກຂອງ loop for ທີ່ຢູ່ຂາງໃນ (inner loop) ແລະ ຈະເຮັດວຽກພາຍໃນ loop for ທີ່ຢູ່ຂາງໃນສຳເລັບສົມບູນຈິ່ງຈະກັບອອກໄປເຮັດວຽກຢູ່ loop for ນອກອີກຄັ້ງ, ເຊິ່ງຈະເຮັດວຽກແບບນີ້ເລື້ອຍໆຈົນກວາເງື່ອນໄຂຂອງ loop ນອກເປັນຜິດ.

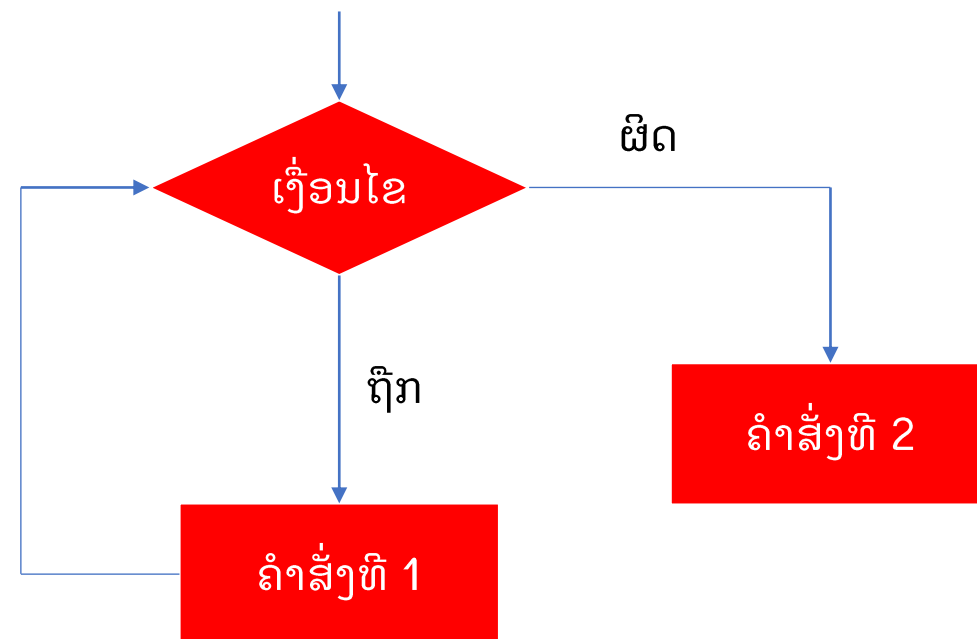
```
example16.c example6.c exampleofforloop.c example7.c
1  #include<stdio.h>
2  #include<stdlib.h>
3  main(){
4      system("color f2");
5      printf("\t===== Example 7 =====\n");
6      int i,j;
7      for(i=1;i<=3;i++){
8          printf("\t i = %d\n",i);
9          for(j=1;j<=i;j++){
10             printf("\t j = %d\n",j);
11         }
12         printf("\n");
13     }
14     printf("\t===== Example 7 =====\n");
15 }
```



II. ຄຳສັ່ງທຳຊ້າ (Iteration Statement).

❑ ຄຳສັ່ງເຮັດຊ້າ while.

```
while(ເງື່ອນໄຂ){  
    ຄຳສັ່ງທີ 1;  
}  
    ຄຳສັ່ງທີ 2;
```





II. ຄຳສັ່ງທຳຊ້າ (Iteration Statement).

❑ ຄຳສັ່ງເຮັດຊ້າ while.

example16.c

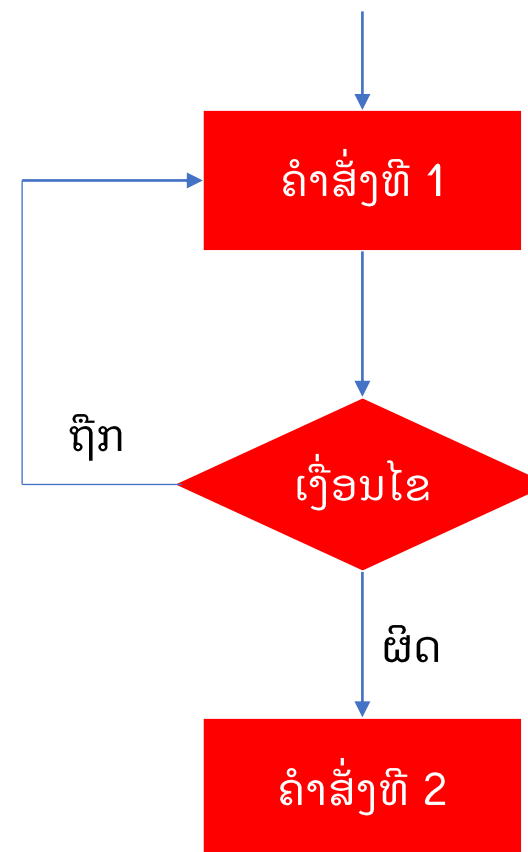
```
1  #include<stdio.h>
2  #include<stdlib.h>
3  main(){
4      printf("\t===== example 16 =====\n");
5      system("color f6");
6      int number;
7      printf("\tGuess number:");
8      scanf("%d",&number);
9      while(number !=3){
10         printf("\tYou are wrong! please try a gain\n");
11         printf("\tThe number is 0-9\n");
12         printf("\tGuess number:");
13         scanf("%d",&number);
14     }
15     printf("\tYou win! Congratulation\n");
16     printf("\t===== example 16 =====\n");
17 }
```



II. ຄຳສັ່ງທຳຊ້າ (Iteration Statement).

❑ ຄຳສັ່ງເຮັດຊ້າ do-while.

```
do{  
    ຄຳສັ່ງທີ 1;  
}while(ເງື່ອນໄຂ);  
ຄຳສັ່ງທີ 2;
```





II. ຄຳສັ່ງທຳຊ້າ (Iteration Statement).

❑ ຄຳສັ່ງເຮັດຊ້າ do-while.

```
example6.c example7.c example16.c example17.c exampleofforloop.c example8.c
1  #include<stdio.h>
2  #include<stdlib.h>
3  main(){
4      printf("\t===== example 16 =====\n");
5      system("color f6");
6      int choice;
7      printf("\t===== \n");
8      printf("\tPlease Select the choice\n");
9      printf("\t===== choice 1 get gun          =====\n");
10     printf("\t===== choice 2 get plane          =====\n");
11     printf("\t===== choice 3 get car          =====\n");
12     printf("\t===== choice 4 get explosives =====\n");
13     printf("\t===== choice 5 get life          =====\n");
14     printf("\t===== \n");
15     do{
16         printf("\n\tSelect(Please select the choice 1-5):");
17         scanf("%d",&choice);
18     }
19     while(choice<1 || choice>5);
20     printf("\tThe game is over you win\n");
21     printf("\t===== example 16 =====\n");
22 }
```



ຖາມ ແລະ ຕອບຄໍາຖາມ



