CSE- 105
Structure Programming

Lecture 9

Loop : break, continue and exit

Loop

```c
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;

for (int k=1; k<=5; k++)
{
    scanf("%lf",&x);
    if (x > 10.0)

        sum += x;
}

printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

```
1
2
3
14
15
Sum = 29.000000
Good Bye
Press any key to continue_
```

break statement

```
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;

for (int k=1; k<=5; k++)
{
    scanf("%lf",&x);
    if (x > 10.0)
        break ;
        sum += x;
}

printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

break;

terminates loop
execution continues with the first statement following the loop

```
1
2
3
14
Sum = 6.000000
Good Bye
Press any key to continue_
```

If the condition is true
Jump out of the loop

break statement

```
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;

for (int k=1; k<=5; k++)
{
     scanf("%lf",&x);
     if (x > 10.0)
     {
          break ;
          sum += x;
     }
}
printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

```
1
2
3
14
Sum = 0.000000
Good Bye
Press any key to continue
```

sum += x;
will never execute, WHY?

continue statement

```
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;

for (int k=1; k<=5; k++)
{
     scanf("%lf",&x);
     if (x > 10.0)
        continue ;
        sum += x;
}

printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

continue;

forces next iteration of the loop, skipping any remaining statements in the loop

If the condition is true go to the next iteration

```
1
2
3
14
15
Sum = 6.000000
Good Bye
Press any key to continue
```

continue statement

```
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;

for (int k=1; k<=5; k++)
{
    scanf("%lf",&x);
    if (x > 10.0)
    {  continue ;
       sum += x;
    }
}
printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

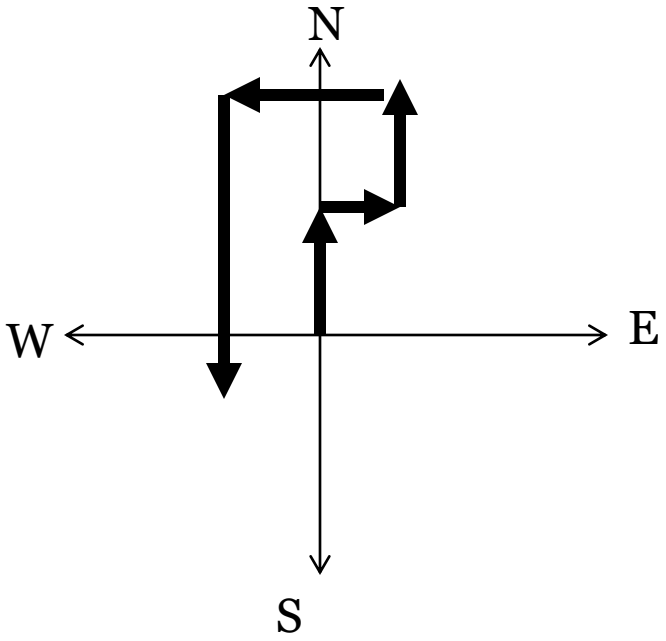
```
1
2
3
14
15
Sum = 0.000000
Good Bye
Press any key to continue_
```

sum += x;

Also never execute here, WHY?

Example: A man walks

- Suppose a man (say, A) stands at (0, 0) and waits for user to give him the direction and distance to go.
- User may enter N E W S for north, east, west, south, and any value for distance.
- When user enters 0 as direction, stop and print out the location where the man stopped



```
float x=0, y=0;
 char direction;
 float mile;
 while (1) {
      printf("Please input the direction as N,S,E,W (0 to exit): ");
      scanf("%c", &direction);          fflush(stdin);
      if (direction=='0'){  /*stop input, get out of the loop */
        break;
      }
      if (direction!='N' && direction!='S' && direction!='E' && direction!='W') {
        printf("Invalid direction, re-enter \n");
        continue;
      }
      printf("Please input the mile in %c direction: ", direction);
      scanf ("%f",&mile);    fflush(stdin);
      if (direction == 'N'){                    /*in north, compute the y*/
        y+=mile;
      } else if (direction == 'E'){             /*in east, compute the x*/
        x+=mile;
      } else if (direction == 'W'){             /*in west, compute the x*/
        x-=mile;
      } else if (direction == 'S'){             /*in south, compute the y*/
        y-=mile;
      }
 }
 printf("\nCurrent position of A: (%4.2f,%4.2f)\n",x,y);       /* output A's location */
```

Example: what will be the output

```
int main()
{
  int a, b, c;
  a=5;
  while(a > 2) {
    for (b = a ; b < 2 * a ; b++ ) {
        c = a + b;
        if (c < 8) continue;
        if (c > 11) break;
        printf( "a = %d   b = %d    c = %d \n", a, b, c);
    } /* end of for-loop */
    a--;
  } /* end of while loop */
}
```

a = 5 b = 5 c = 10
a = 5 b = 6 c = 11

a = 4 b = 4 c = 8
a = 4 b = 5 c = 9

a = 4 b = 6 c = 10
a = 4 b = 7 c = 11

a = 3 b = 5 c = 8

goto statement

```
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;
Z:
for (int k=1; k<=5; k++)
{
     scanf("%lf",&x);
     if (x > 10.0)
      goto Z;
      sum += x;
}

printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

If the condition is true goto label Z

```
1
2
3
14
15
1
2
3
4
5
Sum = 21.000000
Good Bye
Press any key to continue
```

goto statement

```
#include<stdio.h>
int main(void)
{
double sum, x;
sum = 0;

for (int k=1; k<=5; k++)
{
    scanf("%lf",&x);
    if (x > 10.0)
     goto Z;
     sum += x;
}
Z:
printf("Sum = %f \n",sum);
printf("Good Bye\n");

return 0;
}
```

If the condition is true goto label Z,
Which is now actually out of loop

```
1
2
3
14
Sum = 6.000000
Good Bye
Press any key to continue
```

Exercise

- Design the following code using goto and if-else only, i.e., no while or for loop

```
#include<stdio.h>
int main(void)
{

int x = 1;
int i = 1;

while (i <= 9) {
    x = x * i;
    i = i + 1;

    printf("%d    %d \n",x, i);
}

printf("Good Bye\n");

return 0;
}
```

```
1           2
2           3
6           4
24          5
120         6
720         7
5040        8
40320       9
362880      10
Good Bye
Press any key to continue_
```

Exercise

- Design the following code using goto and if-else only, i.e., no while or for loop

```c
#include<stdio.h>
int main(void)
{

int x = 1;
int i = 1;

while (i <= 9) {
    x = x * i;
    i = i + 1;

    printf("%d   %d \n",x, i);
}

printf("Good Bye\n");

return 0;
}
```

```c
#include<stdio.h>
int main(void)
{

int x = 1;
int i = 1;

Z:
if (i <= 9) {
    x = x * i;
    i = i + 1;
    printf("%d   %d \n",x, i);
    goto Z;
}
printf("Good Bye\n");

return 0;
}
```

Exercise

- Design the following code using goto and if-else only, i.e., no while or for loop

```
1          2
2          3
6          4
24         5
120        6
720        7
5040       8
40320      9
362880     10
Good Bye
Press any key to continue_
```

```c
#include<stdio.h>
int main(void)
{

int x = 1;
int i = 1;

Z:
if (i <= 9) {
     x = x * i;
     i = i + 1;
     printf("%d    %d \n",x, i);
     goto Z;
}
printf("Good Bye\n");

return 0;
}
```

Exercise

- What will be the output now ?

```
1          2
Good Bye
Press any key to continue_
```

```
#include<stdio.h>
int main(void)
{

int x = 1;
int i = 1;

if(i <= 9) {
     x = x * i;
     i = i + 1;

     printf("%d    %d \n",x, i);
     goto Z;
}

Z:
printf("Good Bye\n");

return 0;
}
```

Cautions

- Avoid goto as much as possible
  - Reduce Programme readability
  - use COMMENTS, if goto is extremely required

    // or //**

  - For the termination you can also use exit() functions
    which requires a #include<stdlib.h>
    exit(0) → normal programme terminations

Summary

- It will end our loop lectures.
- There will two labs on it lab 4 and 5.
- However you can’t solve any problem in the up-comming labs (Lab 3-11) without loop