Special way to write loop

C_38 ⇒ While loop in C (Part 2)

* We can write while loop in many ways also.

general syntax of while loop:

```
Initialization;
While( Condition)
 {
  // body;
  update/modify;
 }
```

Program 1 (general)

```
void main()
{
  int i=1;
  while(i<10)
  {
   printf ("%d\n", i);
    i++;
  }
   getch();
}
```

O/P:

```
1
2
3
;
;
;
9
```

i

~~1~~ 2

Program 2:

```
void main()
{
  int i=1;
  while(i==1)
  {
   printf ("%d\n", i);
    ++i;
  }
  printf (" End of pgm");
  getch();
}
```

O/P:

```
1
End of pgm.
```

i

~~1~~ ~~2~~.~~1~~ ~~2~~1 ------

Program 3:

```
int i=1;
  while (i=1) -> not condition; assignment
  {
  printf("%d\n", i).
   ++i;
  } printf ("End of pgm").
   getch();
}
```

O/P:

```
1
1
!
!
Infinite loop.
```

Hint

while(1)
↓
True

while(0)
↓
False

i

~~10~~ 11 12 ... 32767 -32767 ...

Program 4 :-

```
int i=10;
while (1)
{
  printf("%d\n", i);
   ++i;
 } printf("End of pgm");
  getch();
}
```

while(1)
↓
True

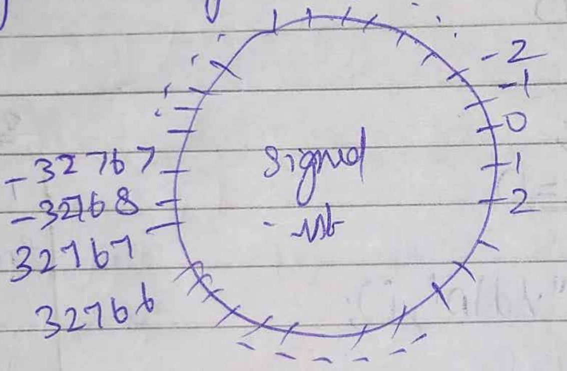
while(0)
↓
False.

O/P:

```
10
11
12
:
:
Infinite loop
```

Hint → O/P

* Infinite loop only; but prints within integer range (i.e) -32768 to -32767



Infinite loop

O/P same

```
10
11
!

32767
-32768
-32767
!
Infinite loop
like circle
```

Program 5 :-

```
int i=10;
while(i)
{
  printf("%d\n", i);
  ++i;
}printf("End of pgm");
getch(); printf("i=%d", i);
}.
```

i

~~10~~
11
:
32767
-32768
:
-1
i → 0

while(0)
↓
false

loop terminate

O/P:

```
10
11
:
:
32767
-32768
-1
:
i=0
End of pgm
```

i
[10]

Program 6:-

```
 int i=10;
while (i)
{

  printf("%d\n", i).
 }
printf("End of pgm");
getch();    printf("i = %d", i);
}
```

O/P :-

```
10
10
10
:
:
Infinite loop
```

* Here no modify/update; so it is infinite loop.

Program 7:-

```
 int i;
while (i<=10)
{
  printf("%d\n", i);
   i++;
 }
printf(" End of pgm");
getch();  printf("i = %d", i);
}
```

i
[~~0~~]
~~1~~
~~2~~
:
~~10~~
11

O/P

```
0
1
2
3
4
5
:
10
End of pgm.
i = 11
```

Program 8:-

```
 int i=1;
while (____)  no condition
{
printf("%d\n", i);
 i++;
} printf("End of pgm");
getch();
}
```

Note:
* While ( ) Cannot be blank without Condition it gives error.

i
[1]

O/P

```
Error.
```

(*) **Hint** while(condition) → Condition can be (initialization) or (expression) or (condition) or (modification) [while(i++) is expression]

i: ~~0~~ 1

while(0)

Program 9 :-

```
int i = 0;
while (i++)
{
 printf("%d\n", i);
}
printf ("End of pgm");
getch(); printf("i=%d\n", i);
}
```

O/P :-

```
End of pgm
i = 1.
```

```
     expression
while (i ++)  -i++->  post increment   | 0 ++ ⇒ 1
while (0) ⇒ false
  {
   printf("%d\n", i);
  }
printf ("End of pgm");
printf(" i = %d\n", i);
getch();
}
```

i: ~~0~~ 1

⇒ **Hint :-** loop terminates at the start itself; it won't enter inside loop

Program 10 :-

```
int i = 0;
while (++i)
{
 printf ("%d\n", i);
}
printf (" End of program");
getch(); printf ("i%d\n", i);
}
```

i: ~~0~~ ~~1~~ 2 .... 32768 → ~~1~~ (0) loopter

```
while(1)
 while(2)
   :
 while(-32768)
 while(-1)
 while(0)
```

O/P :-

```
1
2
|
-32768
|
-1
loop terminate
End of pgm
i=0
```

Hint → ASCII Value ['a' = 97, 'b' = 98, ...]

Program 11:

```
void main()
{
    char ch = 'a';
    while (ch)
    {
        printf ("%d", ch);
        ch++;
    } getch();   printf ("i=%d", ch);
}
```

ch

a

~~97~~

~~98~~ ....

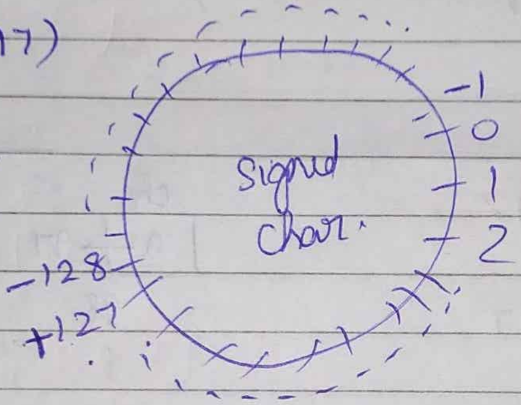
~~-1~~

0

O/P

```
97
98
⋮
127
-128
-127
-1 → loop termi
ch = 0
```

```
→ while (97)
  {
    - - - -
    ch++;
  }
while (98)
while (99)
   ⋮
while (+127)
while (-128)
while (-127)
   ⋮
while (-1)
while (0) → loop terminates
```



Program 12:

```
char ch = 'a';
while (ch)
{
    printf ("%c", ch);
    ch++;
}
printf ("i=%d", ch);
}
```

ch

a

97 → 'a'

98 → 'b'

⋮

~~-1~~ → ''

0

O/P

```
a - 97
b - 98
c
⋮
ch = 0
```

Program 13:

```
Void main()
{
 Char ch='a';
 while(ch)
 {
  printf("%d\n", ch);
  ch--;
 }
  printf("ch=%d\n", ch);
getch();
}
```

ch

a → 97

96

95

⋮

+1

0

O/P:

```
97
96
95
⋮
+1 → loop terminates
ch= 0
```

Program 14:

```
Void main()
{
 Char ch='a';
 while (1)
 {
  printf("%d\n", ch);
   ch++;
 }
  printf("ch=%d\n", ch;
}
```

ch

a → 97

98

⋮

+127

-128

-127

⋮

-1

⋮

0

infinite loop

O/P

```
97
98
+127
-128
-127
⋮
-1
0
infinite loop
```

Program15:

```
Void main()
{
 Char ch='a';
 while (0)
 {
  printf ("%d\n", ch);
  ch++;
 }
} printf ("ch=%d\n", ch);
```

ch

a → 97

while(0)

↓

false

loop terminates

It won't enter loop

O/P:

```
ch= 97
```

int   char ch
[0]   [a] → 97

Program 16:-

```
void main()
{
  int i=0;
  char ch='a';
  while(ch<127&& i==1)      [97 && i==1 → 1&&0 → 0]
  {
    printf("%d\n", ch);
     ch--;

   printf("i=%d\n", i);
   printf("ch=%d\n", ch);
}   printf("ch=%c\n", ch);
```

↓
while(0)
↓
false → loop terminats

It won't enter loop

O/P:

```
i=0
ch=97.
ch=a.
```

Program 17:-

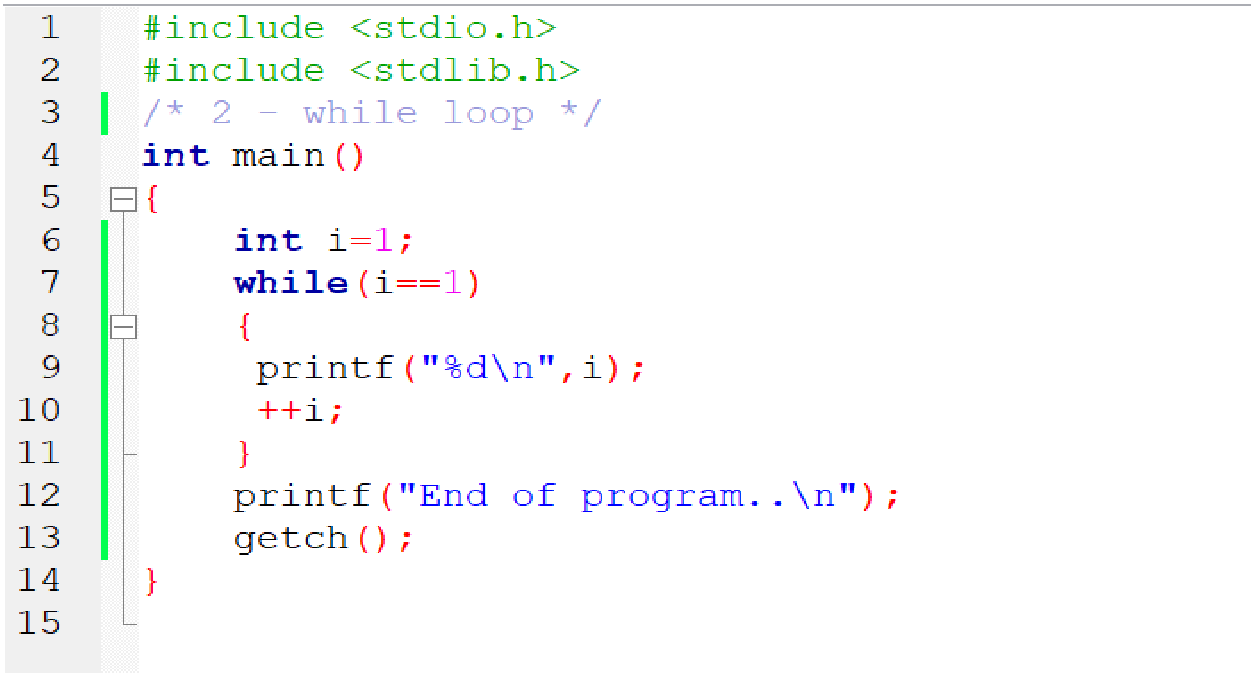
```
void main()
{
 int i=1;
 while(i<=10) ;
 {
   printf("%d\n", i);
  }
  printf("End of program..\n");
   i++;
  printf("i=%d", i);
  getch();
}
```

O/P:

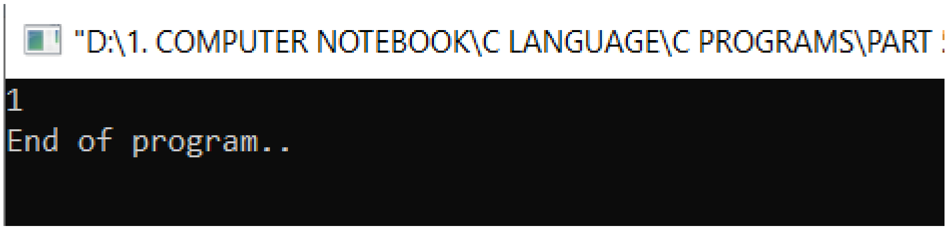
No output.

* While condition with semicolon gives no output because it will be treated as a statement and while(i<=10); → It can't come out with terminate condition; so we get no O/P.

CODE 2:

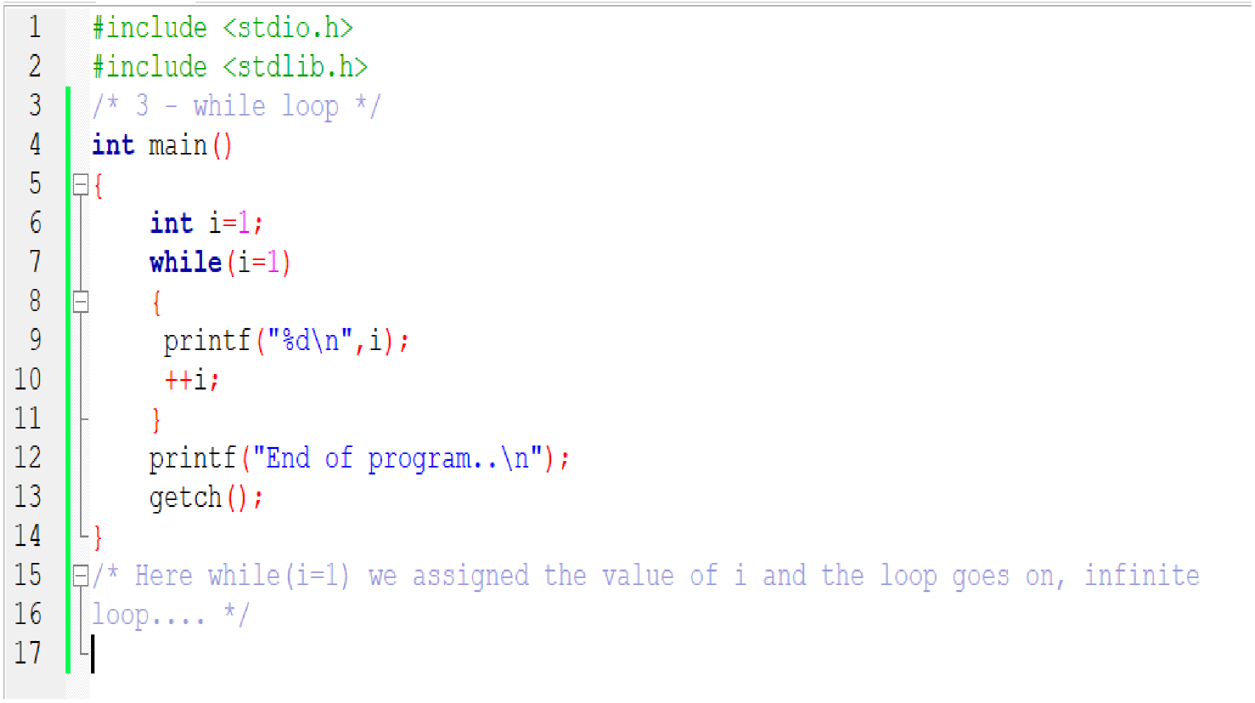


```c
#include <stdio.h>
#include <stdlib.h>
/* 2 - while loop */
int main()
{
    int i=1;
    while(i==1)
    {
     printf("%d\n",i);
     ++i;
    }
    printf("End of program..\n");
    getch();
}
```

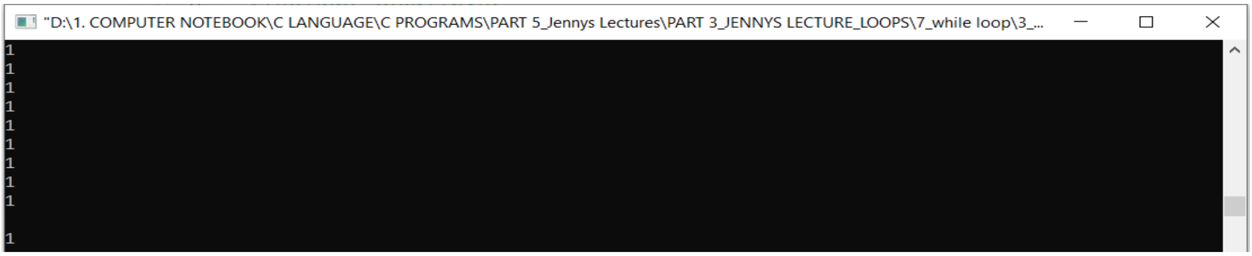


```
1
End of program..
```

CODE 3:



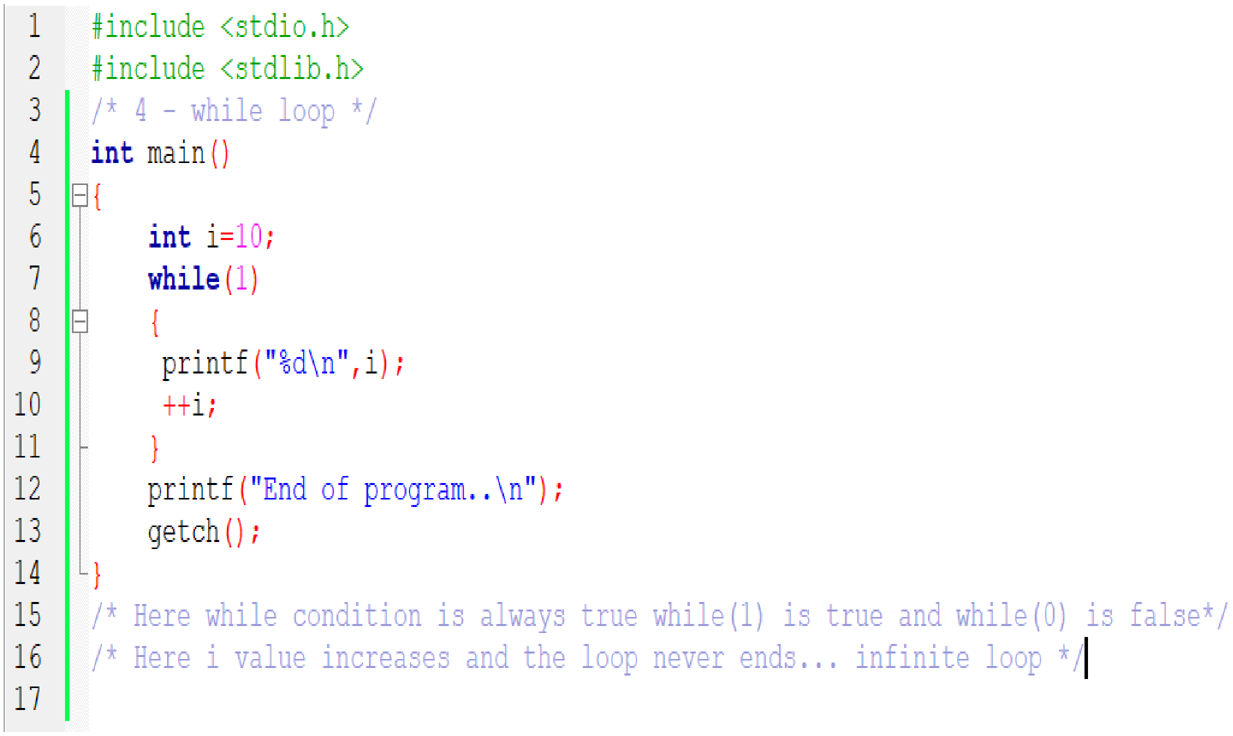
```c
#include <stdio.h>
#include <stdlib.h>
/* 3 - while loop */
int main()
{
    int i=1;
    while(i=1)
    {
     printf("%d\n",i);
     ++i;
    }
    printf("End of program..\n");
    getch();
}
/* Here while(i=1) we assigned the value of i and the loop goes on, infinite
loop.... */
```



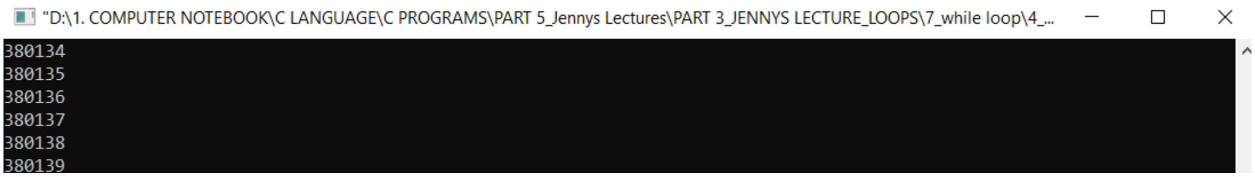
"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_LOOPS\7_while loop\3_...

```
1
1
1
1
1
1
1
1
1

1
```

CODE 4:



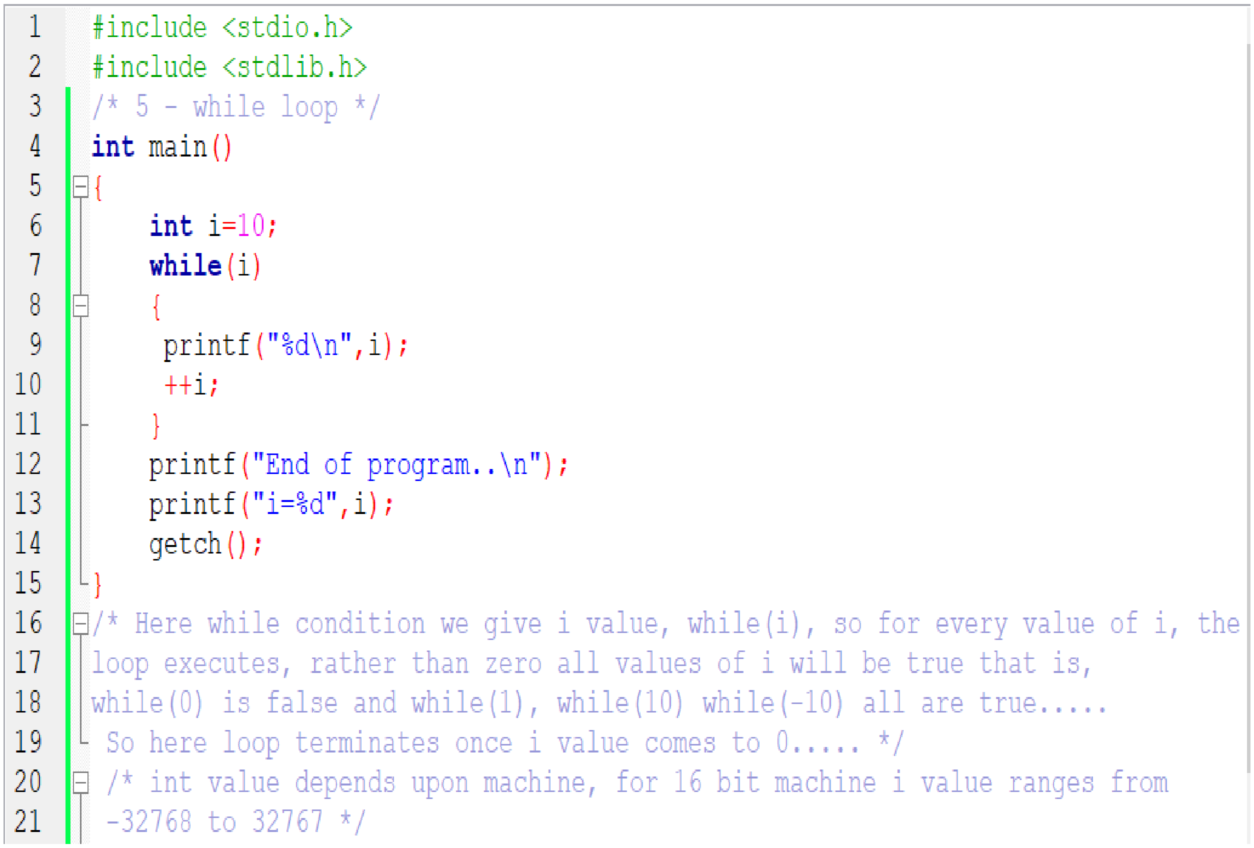
```c
#include <stdio.h>
#include <stdlib.h>
/* 4 - while loop */
int main()
{
    int i=10;
    while(1)
    {
     printf("%d\n",i);
     ++i;
    }
    printf("End of program..\n");
    getch();
}
/* Here while condition is always true while(1) is true and while(0) is false*/
/* Here i value increases and the loop never ends... infinite loop */

```



"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_LOOPS\7_while loop\4_...

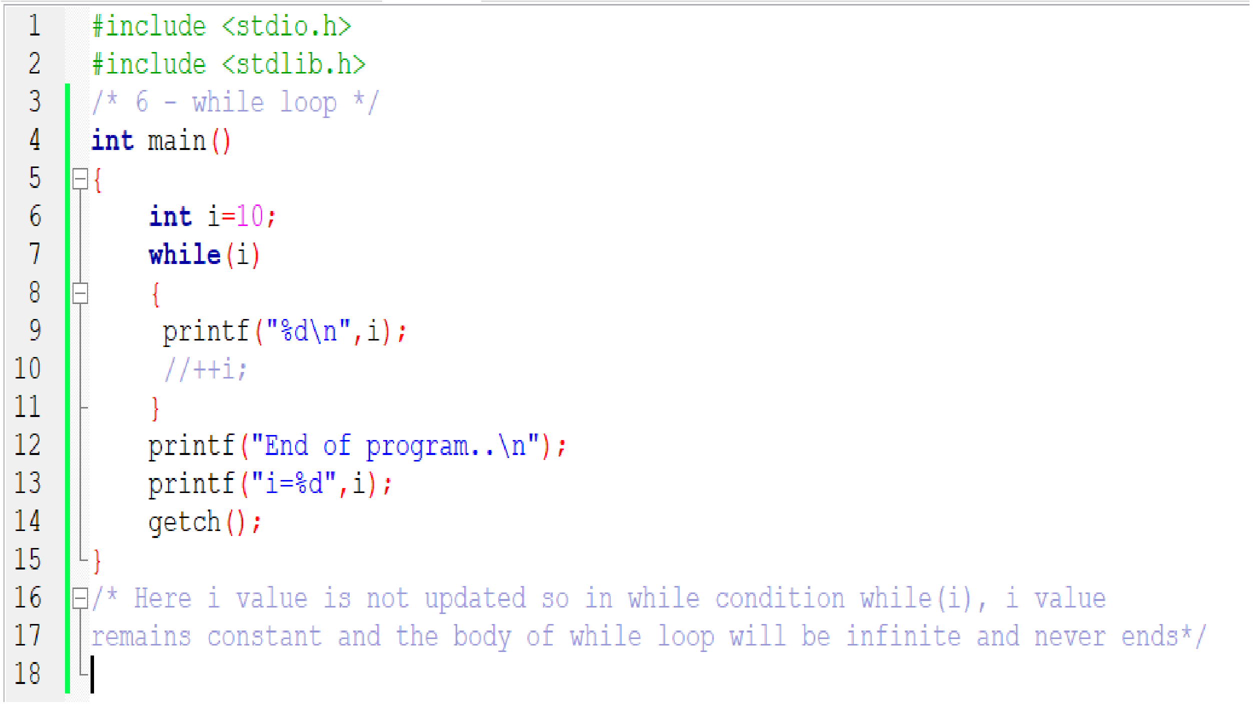
```
380134
380135
380136
380137
380138
380139
```

CODE 5:

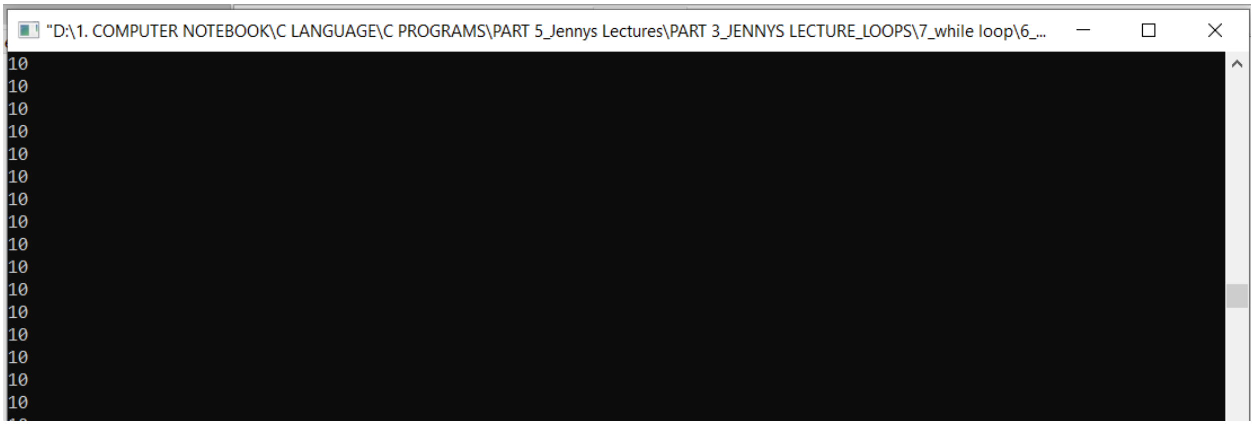


```c
#include <stdio.h>
#include <stdlib.h>
/* 5 - while loop */
int main()
{
    int i=10;
    while(i)
    {
     printf("%d\n",i);
     ++i;
    }
    printf("End of program..\n");
    printf("i=%d",i);
    getch();
}
/* Here while condition we give i value, while(i), so for every value of i, the
loop executes, rather than zero all values of i will be true that is,
while(0) is false and while(1), while(10) while(-10) all are true.....
 So here loop terminates once i value comes to 0..... */
 /* int value depends upon machine, for 16 bit machine i value ranges from
 -32768 to 32767 */
```

CODE 6:



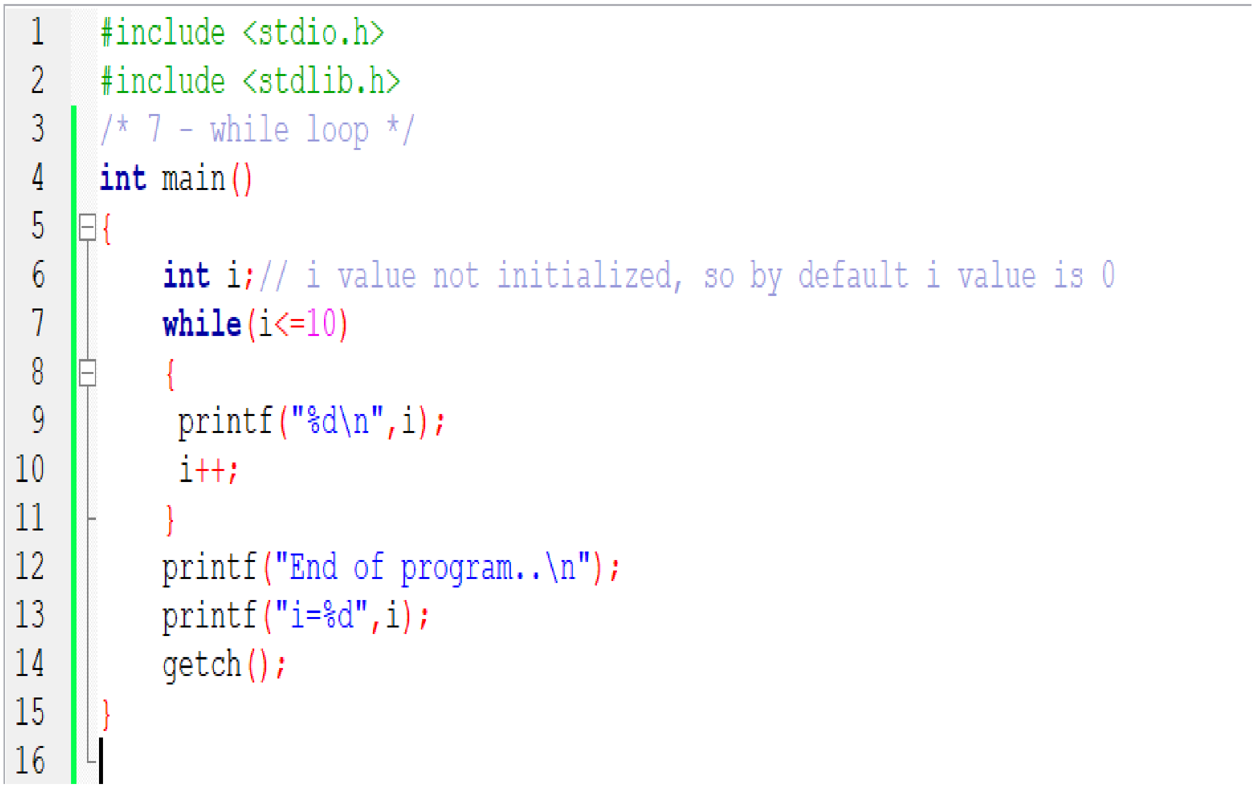
```c
#include <stdio.h>
#include <stdlib.h>
/* 6 - while loop */
int main()
{
    int i=10;
    while(i)
    {
     printf("%d\n",i);
     //++i;
    }
    printf("End of program..\n");
    printf("i=%d",i);
    getch();
}
/* Here i value is not updated so in while condition while(i), i value
remains constant and the body of while loop will be infinite and never ends*/

```



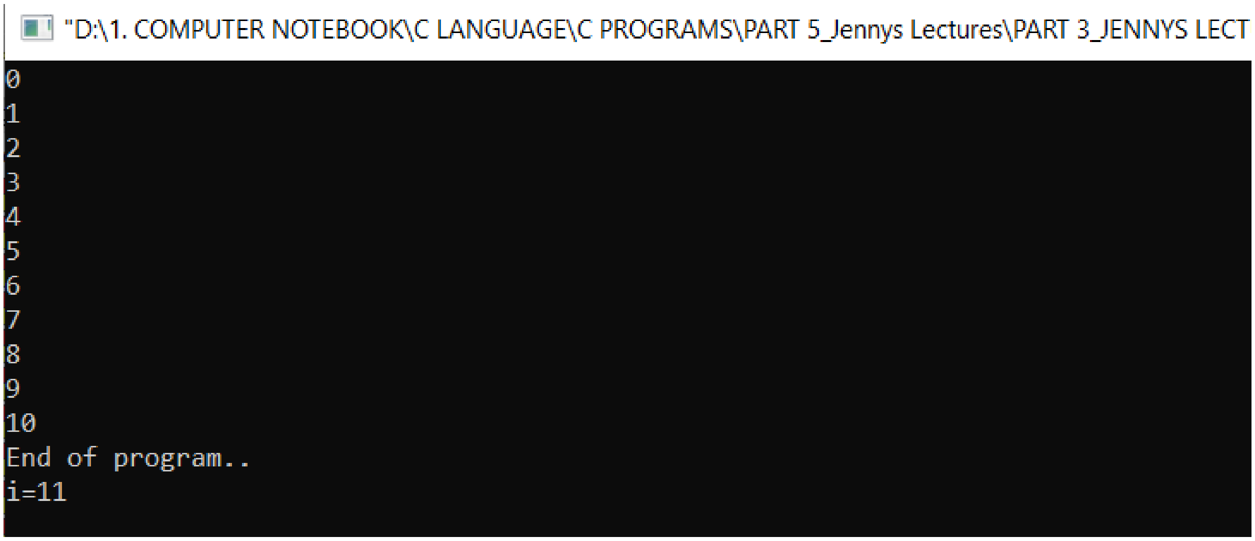
"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_LOOPS\7_while loop\6_...

```
10
10
10
10
10
10
10
10
10
10
10
10
10
10
10
10
```

CODE 7:

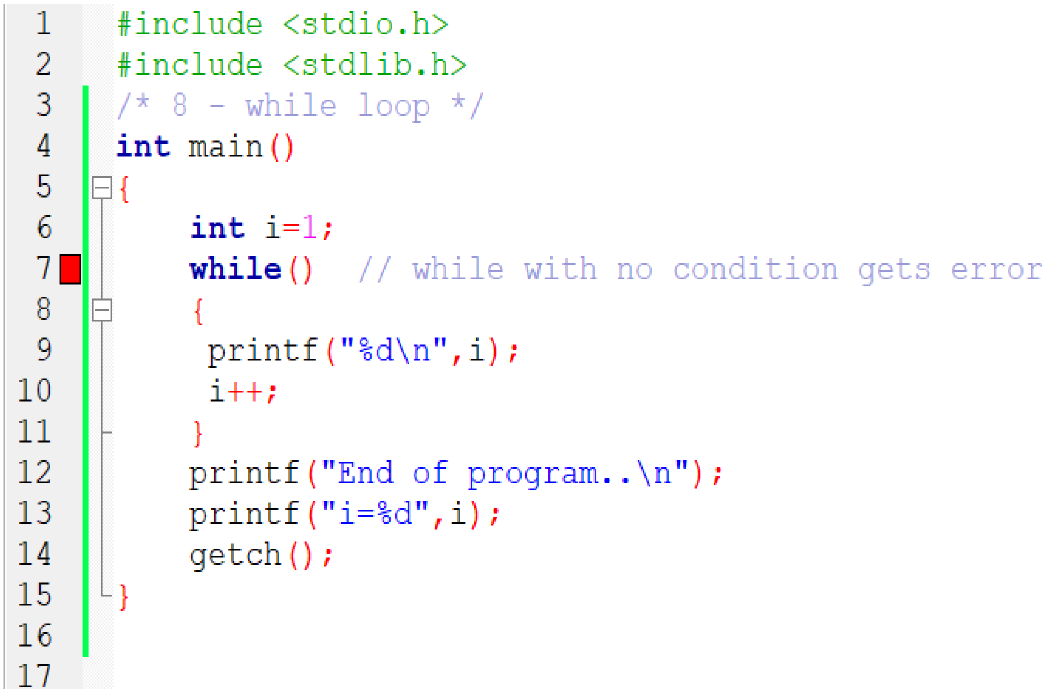


```c
#include <stdio.h>
#include <stdlib.h>
/* 7 - while loop */
int main()
{
    int i;// i value not initialized, so by default i value is 0
    while(i<=10)
    {
     printf("%d\n",i);
     i++;
    }
    printf("End of program..\n");
    printf("i=%d",i);
    getch();
}
```

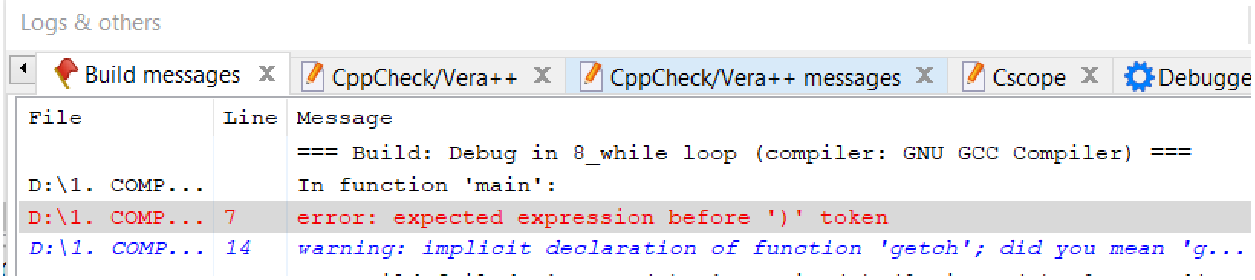

"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECT

```
0
1
2
3
4
5
6
7
8
9
10
End of program..
i=11
```

CODE 8:

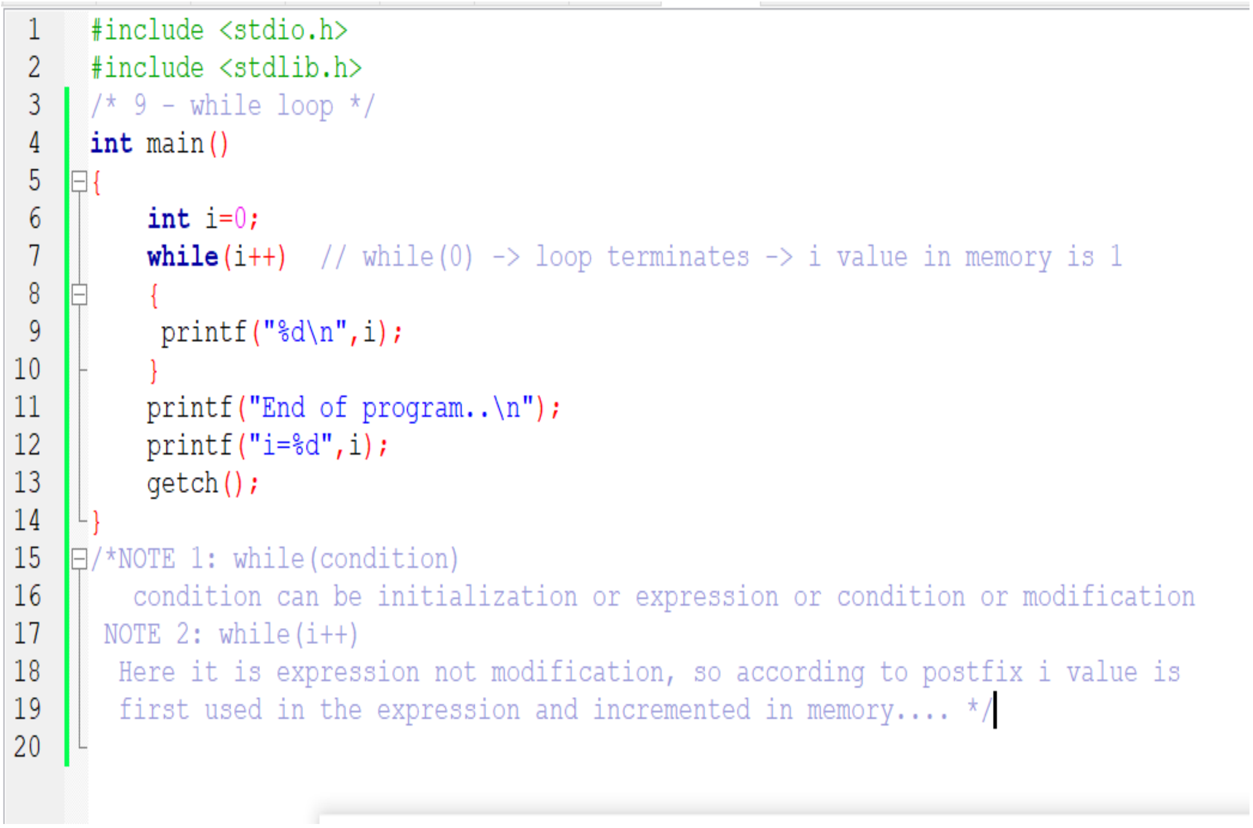


```
#include <stdio.h>
#include <stdlib.h>
/* 8 - while loop */
int main()
{
    int i=1;
    while()  // while with no condition gets error
    {
     printf("%d\n",i);
     i++;
    }
    printf("End of program..\n");
    printf("i=%d",i);
    getch();
}
```

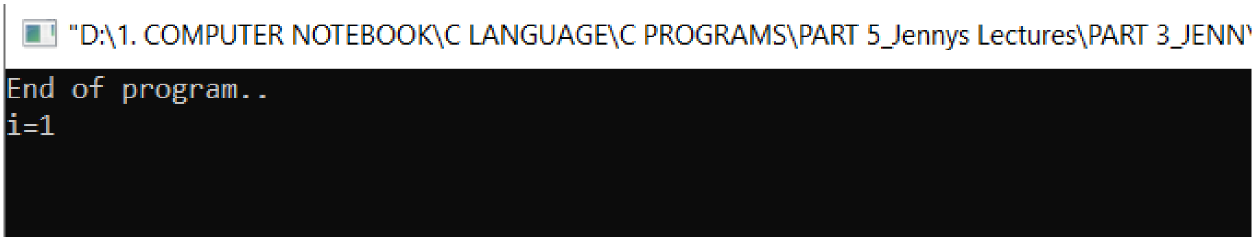


```
File           Line  Message
                     === Build: Debug in 8_while loop (compiler: GNU GCC Compiler) ===
D:\1. COMP...        In function 'main':
D:\1. COMP...  7     error: expected expression before ')' token
D:\1. COMP...  14    warning: implicit declaration of function 'getch'; did you mean 'g...
```

CODE 9:



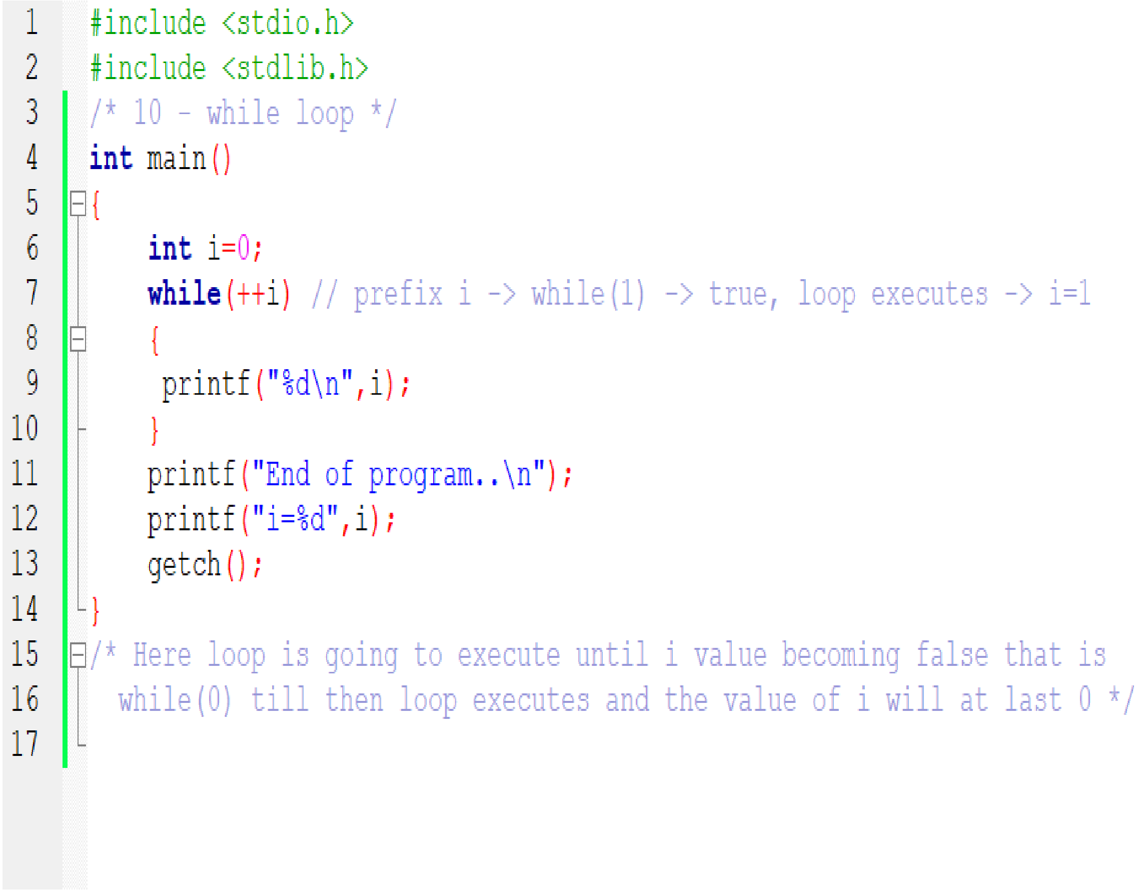
```c
#include <stdio.h>
#include <stdlib.h>
/* 9 - while loop */
int main()
{
    int i=0;
    while(i++)  // while(0) -> loop terminates -> i value in memory is 1
    {
     printf("%d\n",i);
    }
    printf("End of program..\n");
    printf("i=%d",i);
    getch();
}
/*NOTE 1: while(condition)
   condition can be initialization or expression or condition or modification
 NOTE 2: while(i++)
  Here it is expression not modification, so according to postfix i value is
  first used in the expression and incremented in memory.... */
```



"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENN

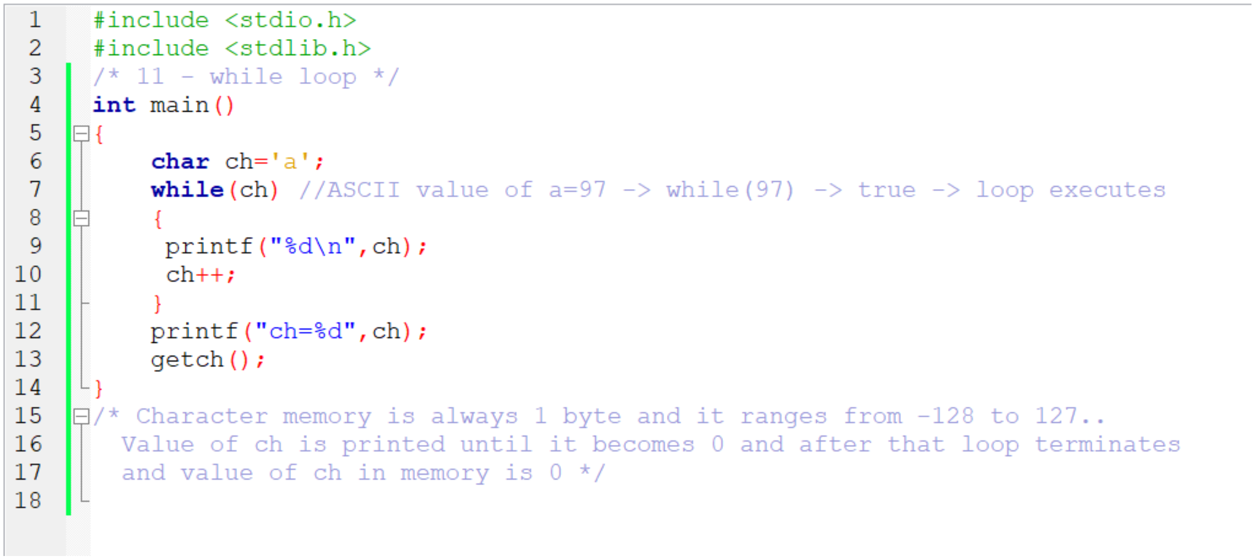
```
End of program..
i=1
```

CODE 10:

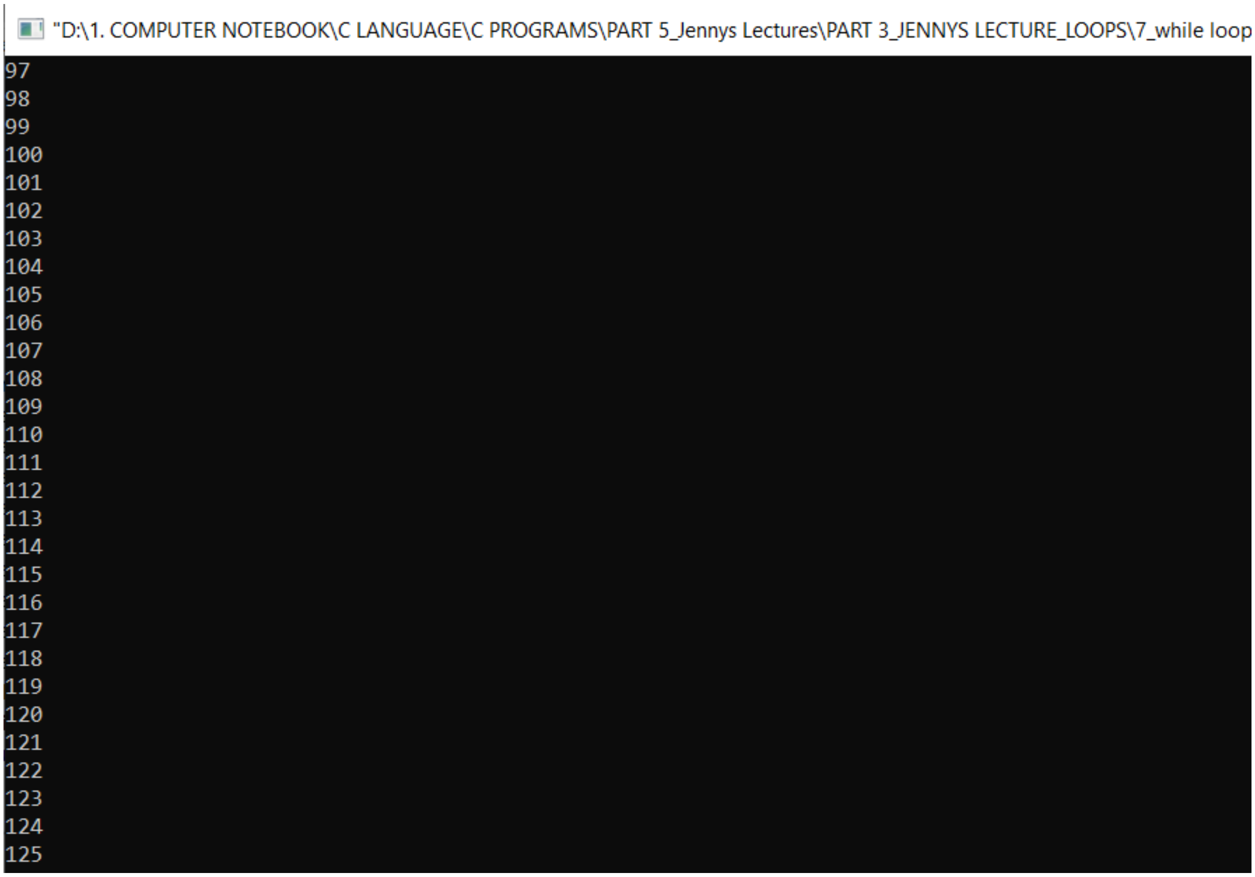


```c
#include <stdio.h>
#include <stdlib.h>
/* 10 - while loop */
int main()
{
    int i=0;
    while(++i) // prefix i -> while(1) -> true, loop executes -> i=1
    {
     printf("%d\n",i);
    }
    printf("End of program..\n");
    printf("i=%d",i);
    getch();
}
/* Here loop is going to execute until i value becoming false that is
  while(0) till then loop executes and the value of i will at last 0 */
```

CODE 11:

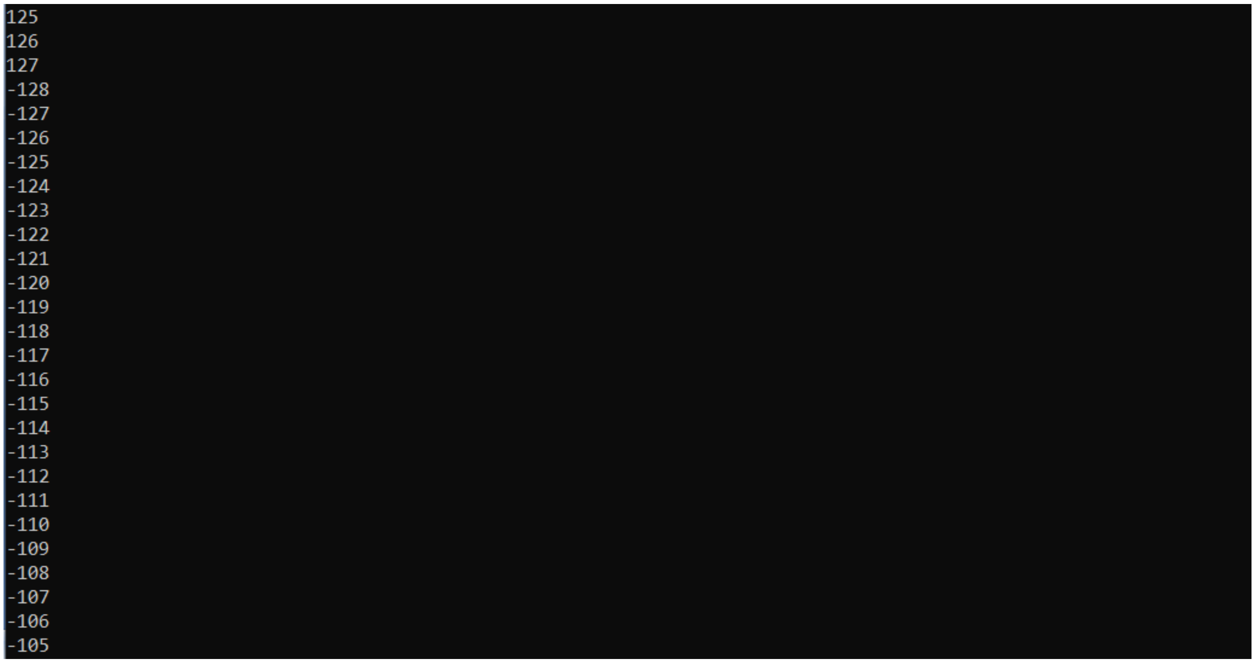


```c
#include <stdio.h>
#include <stdlib.h>
/* 11 - while loop */
int main()
{
    char ch='a';
    while(ch) //ASCII value of a=97 -> while(97) -> true -> loop executes
    {
     printf("%d\n",ch);
     ch++;
    }
    printf("ch=%d",ch);
    getch();
}
/* Character memory is always 1 byte and it ranges from -128 to 127..
  Value of ch is printed until it becomes 0 and after that loop terminates
  and value of ch in memory is 0 */
```

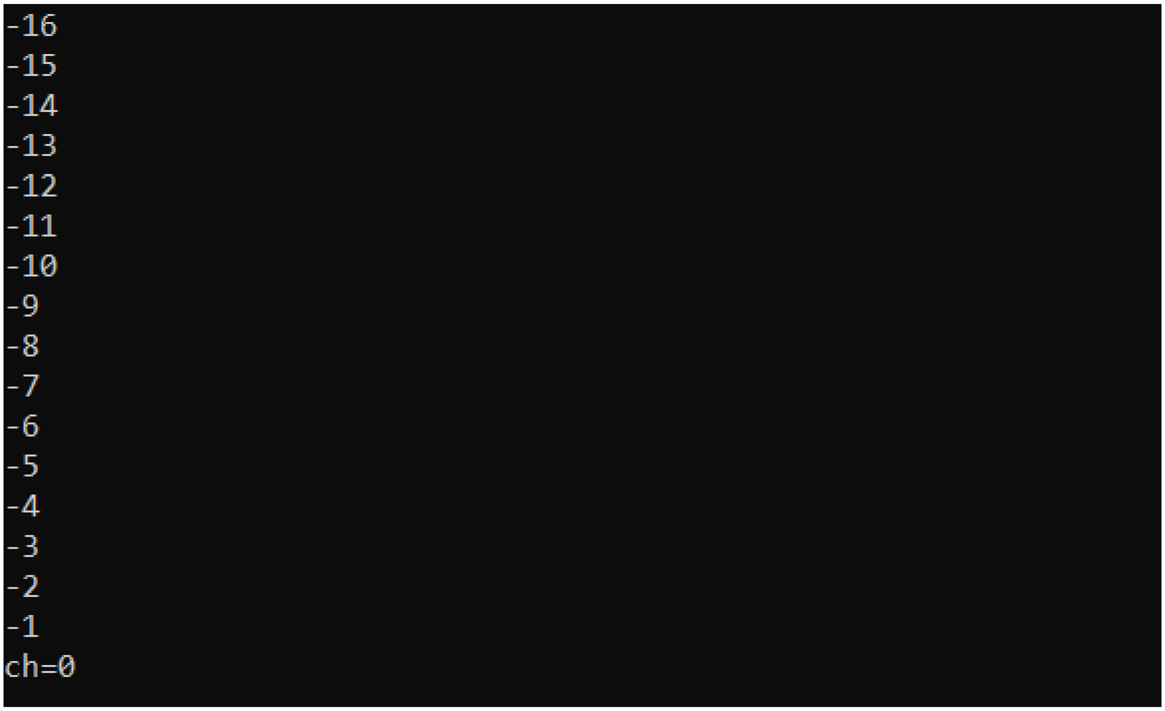


"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_LOOPS\7_while loop

```
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
```

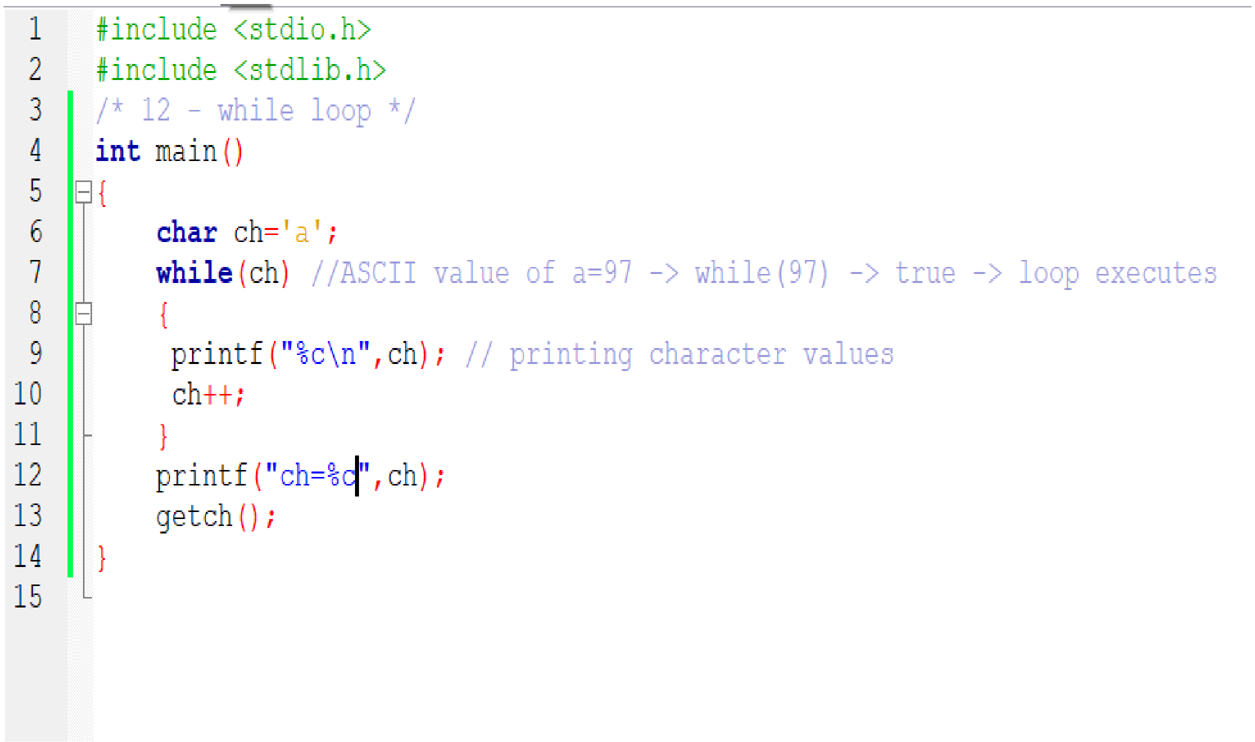


```
125
126
127
-128
-127
-126
-125
-124
-123
-122
-121
-120
-119
-118
-117
-116
-115
-114
-113
-112
-111
-110
-109
-108
-107
-106
-105
```

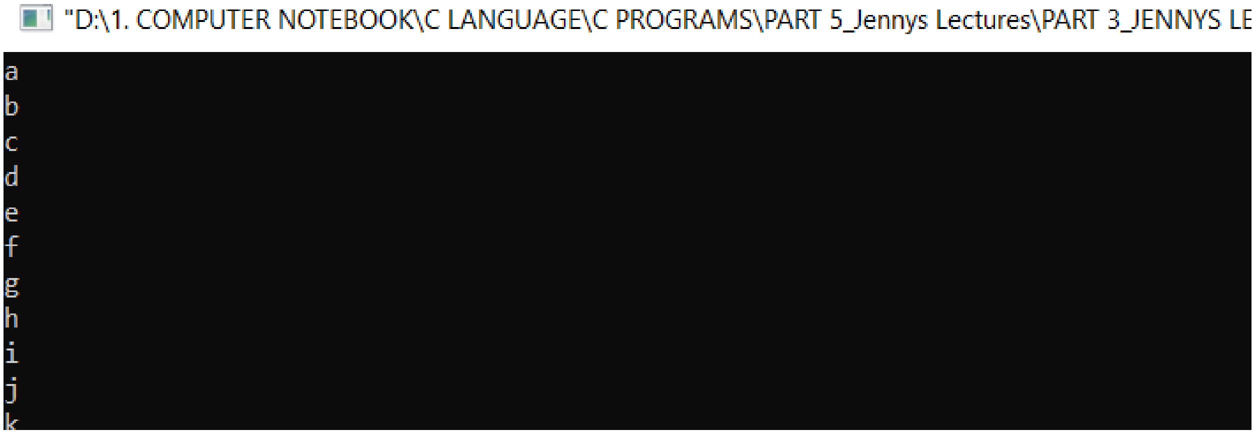


```
-16
-15
-14
-13
-12
-11
-10
-9
-8
-7
-6
-5
-4
-3
-2
-1
ch=0
```

CODE 12:

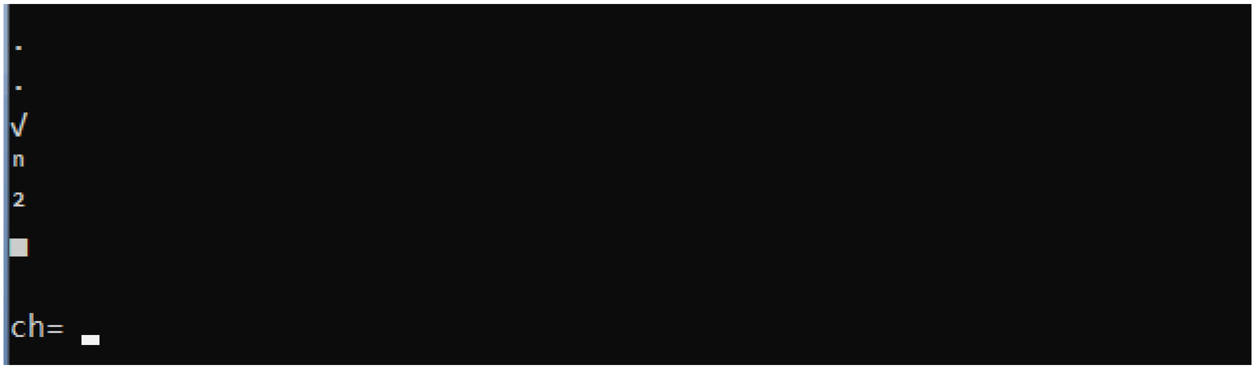


```c
#include <stdio.h>
#include <stdlib.h>
/* 12 - while loop */
int main()
{
    char ch='a';
    while(ch) //ASCII value of a=97 -> while(97) -> true -> loop executes
    {
     printf("%c\n",ch); // printing character values
     ch++;
    }
    printf("ch=%c",ch);
    getch();
}
```



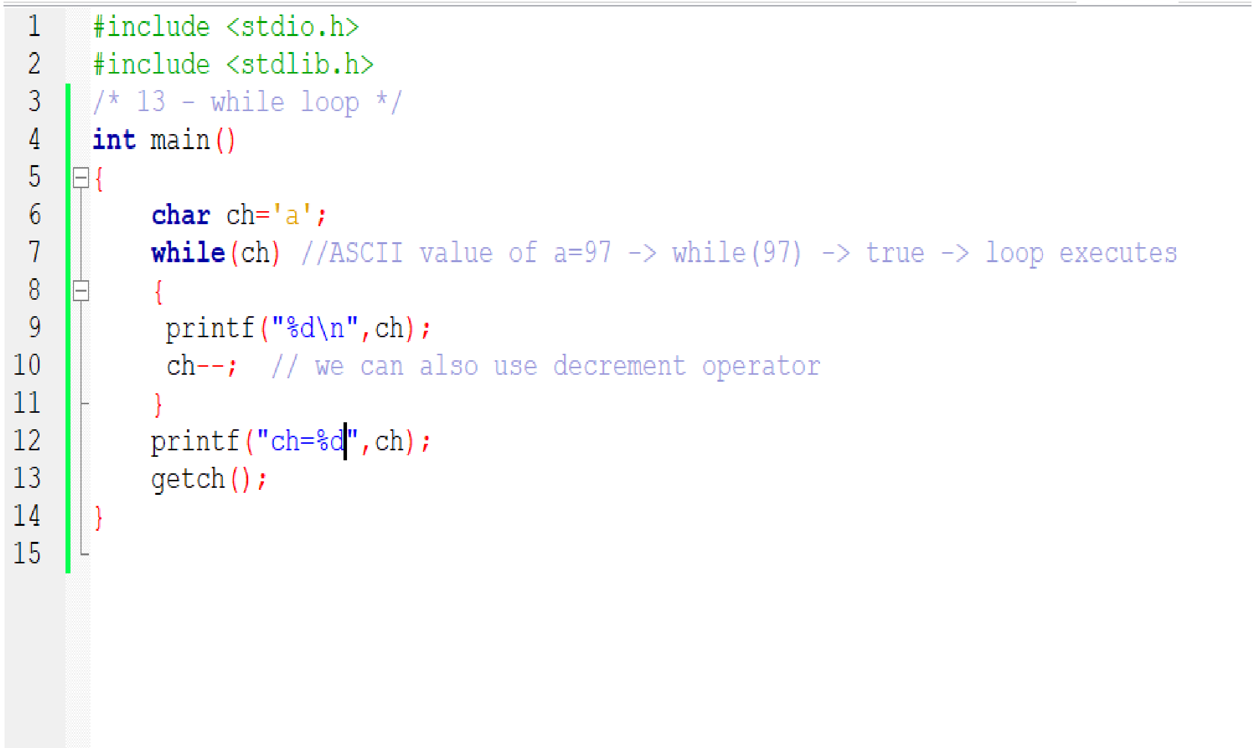
"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LE

```
a
b
c
d
e
f
g
h
i
j
k
```

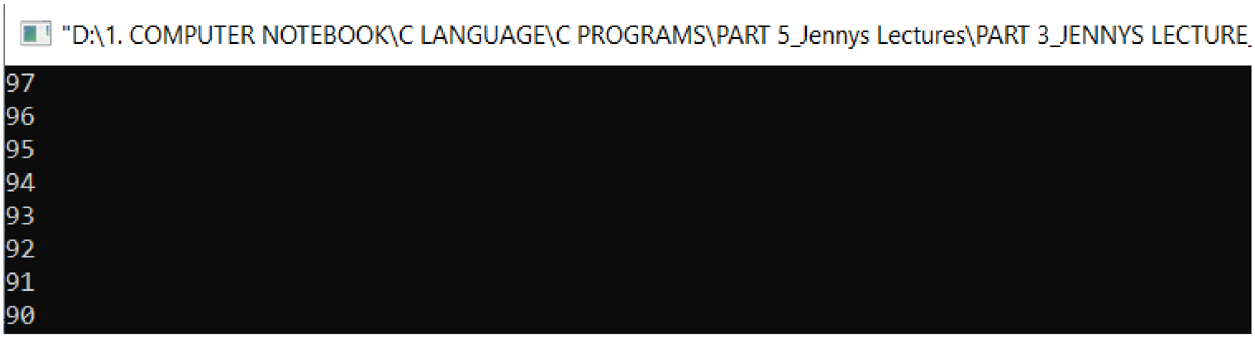


```
·
·
√
ⁿ
²
■

ch= 
```

CODE 13:

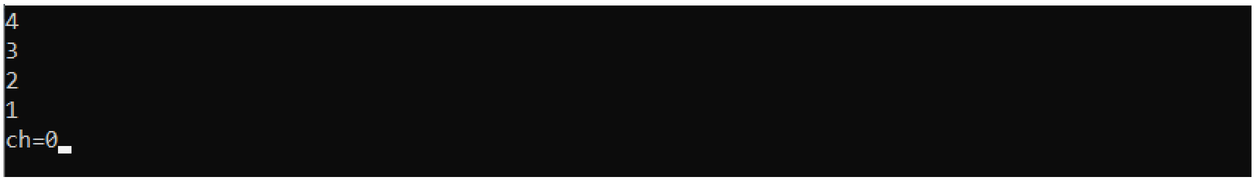


```c
#include <stdio.h>
#include <stdlib.h>
/* 13 - while loop */
int main()
{
    char ch='a';
    while(ch) //ASCII value of a=97 -> while(97) -> true -> loop executes
    {
     printf("%d\n",ch);
     ch--;  // we can also use decrement operator
    }
    printf("ch=%d",ch);
    getch();
}

```



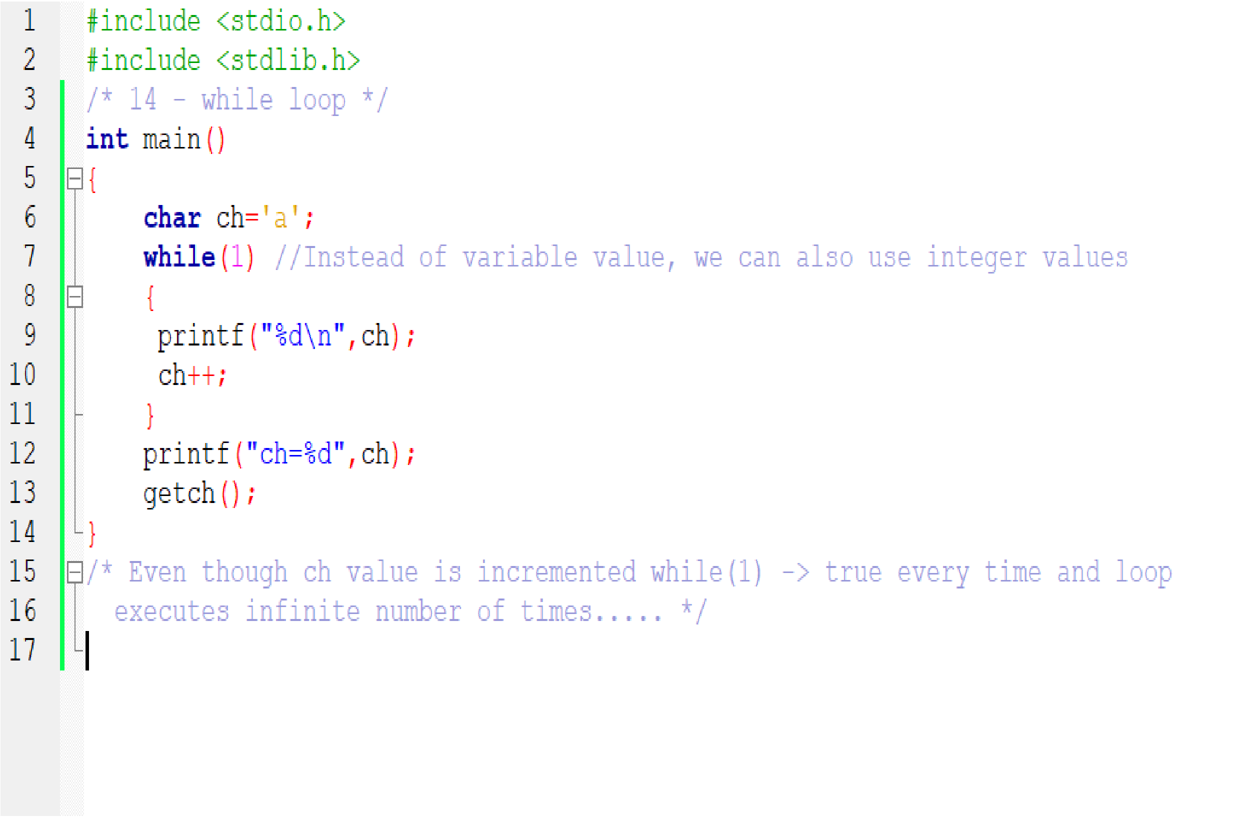
"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_

```
97
96
95
94
93
92
91
90
```

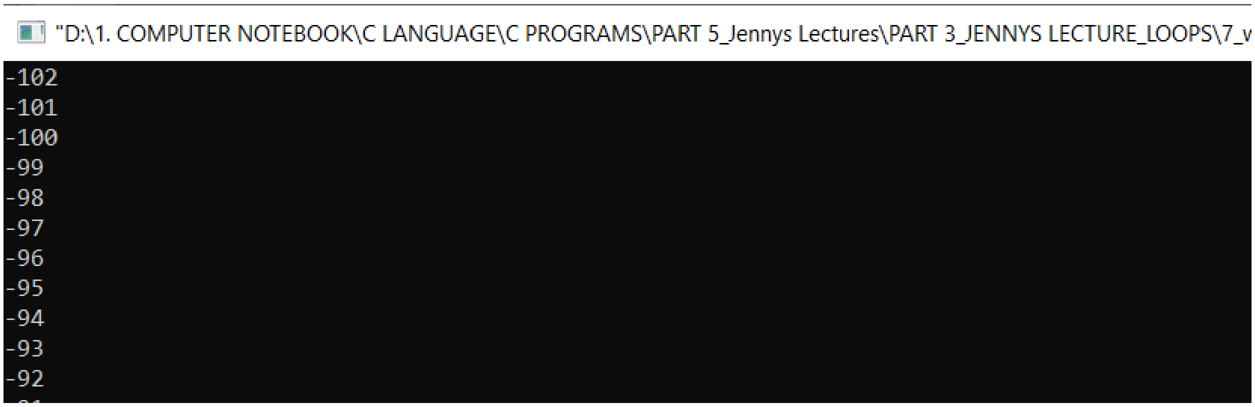


```
4
3
2
1
ch=0_
```

CODE 14:



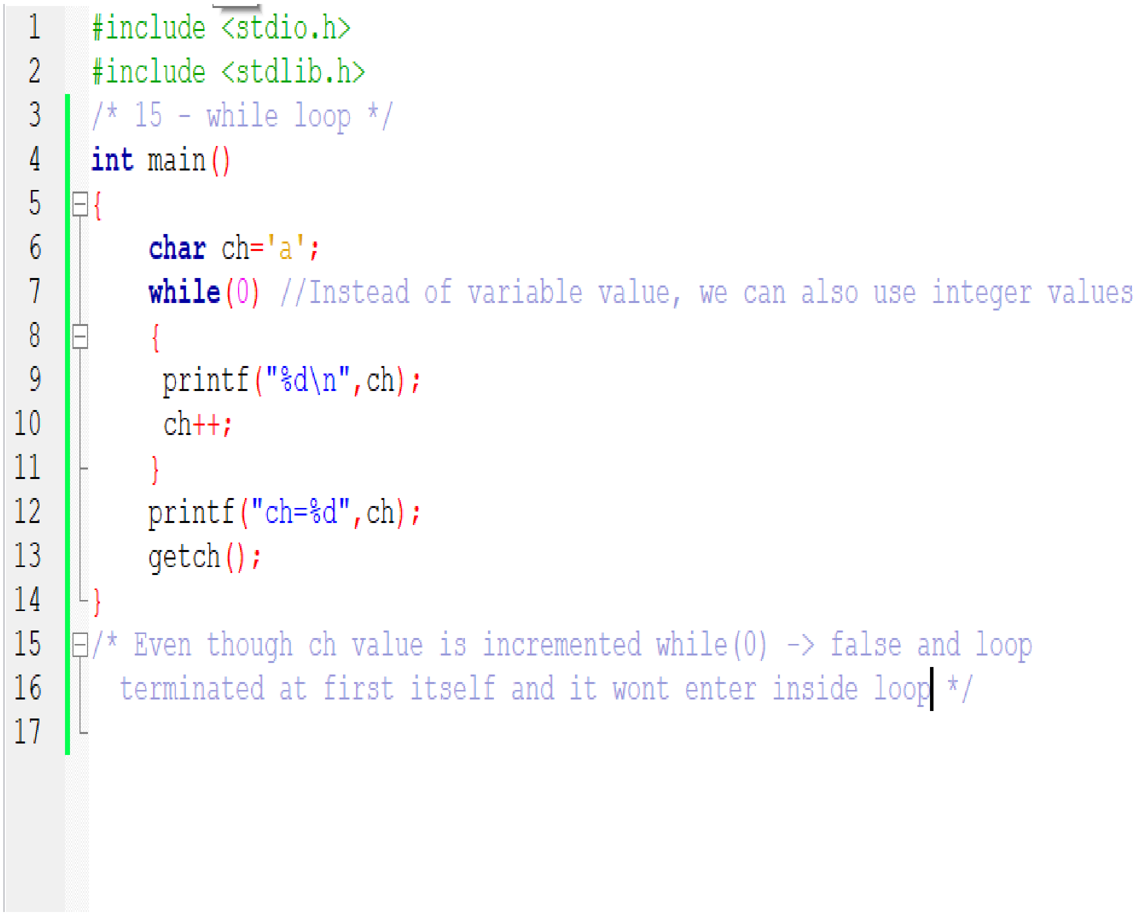
```c
#include <stdio.h>
#include <stdlib.h>
/* 14 - while loop */
int main()
{
    char ch='a';
    while(1) //Instead of variable value, we can also use integer values
    {
     printf("%d\n",ch);
     ch++;
    }
    printf("ch=%d",ch);
    getch();
}
/* Even though ch value is incremented while(1) -> true every time and loop
  executes infinite number of times..... */
```



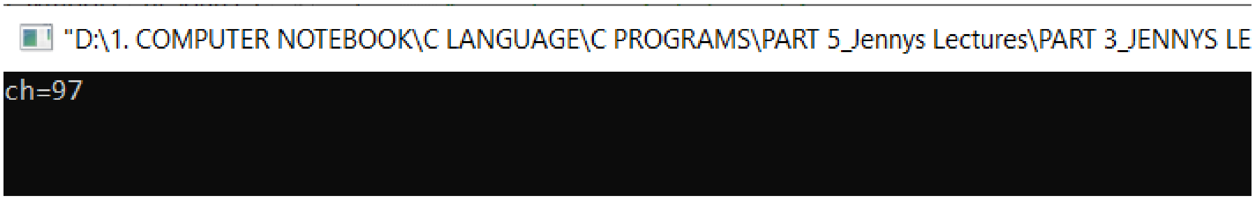
"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_LOOPS\7_v

```
-102
-101
-100
-99
-98
-97
-96
-95
-94
-93
-92
```

CODE 15:



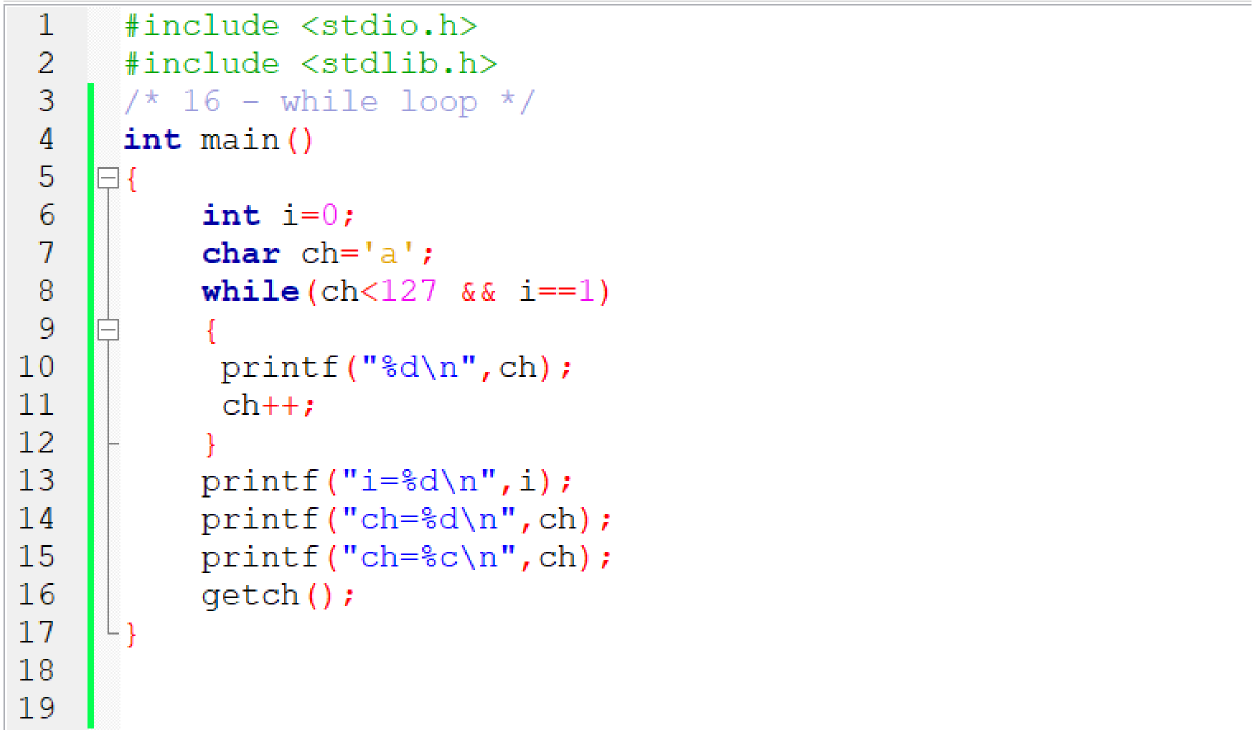
```c
#include <stdio.h>
#include <stdlib.h>
/* 15 - while loop */
int main()
{
    char ch='a';
    while(0) //Instead of variable value, we can also use integer values
    {
     printf("%d\n",ch);
     ch++;
    }
    printf("ch=%d",ch);
    getch();
}
/* Even though ch value is incremented while(0) -> false and loop
  terminated at first itself and it wont enter inside loop */
```



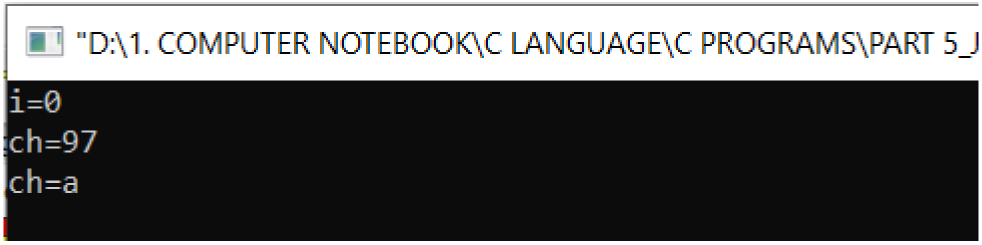
"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LE

```
ch=97
```

CODE 16:

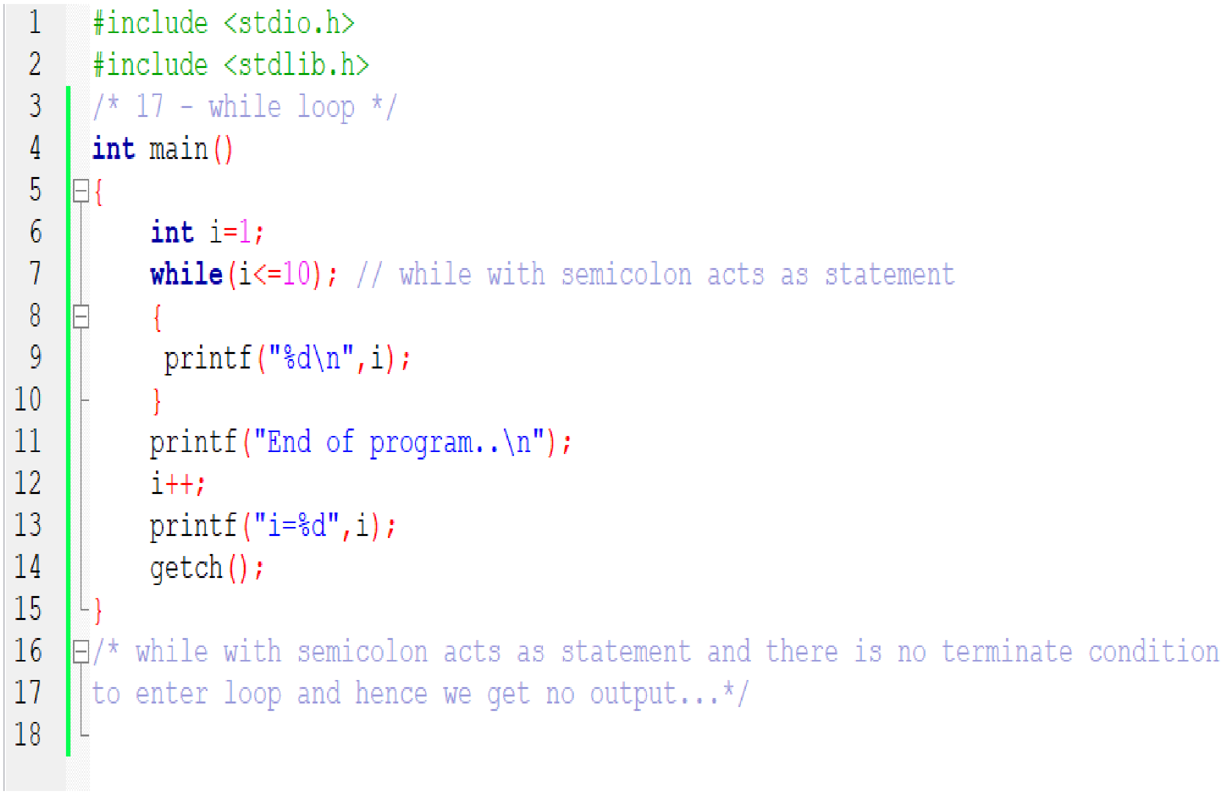


```c
#include <stdio.h>
#include <stdlib.h>
/* 16 - while loop */
int main()
{
    int i=0;
    char ch='a';
    while(ch<127 && i==1)
    {
     printf("%d\n",ch);
     ch++;
    }
    printf("i=%d\n",i);
    printf("ch=%d\n",ch);
    printf("ch=%c\n",ch);
    getch();
}
```

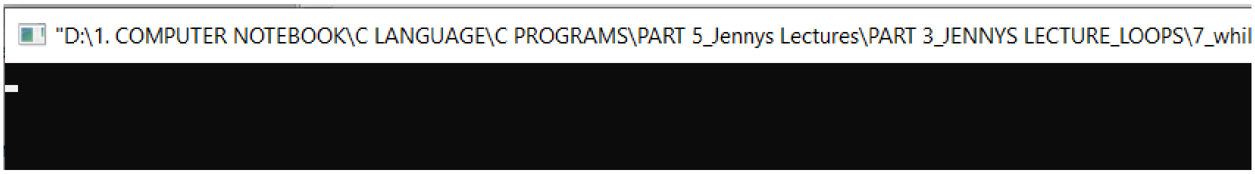


```
i=0
ch=97
ch=a
```

CODE 17:



```
#include <stdio.h>
#include <stdlib.h>
/* 17 - while loop */
int main()
{
    int i=1;
    while(i<=10); // while with semicolon acts as statement
    {
     printf("%d\n",i);
    }
    printf("End of program..\n");
    i++;
    printf("i=%d",i);
    getch();
}
/* while with semicolon acts as statement and there is no terminate condition
to enter loop and hence we get no output...*/
```



"D:\1. COMPUTER NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys Lectures\PART 3_JENNYS LECTURE_LOOPS\7_whil