CSE 113 Structured Programming Language

Control Statement: break, continue & switch

TASNIM ZAHAN
ASSISTANT PROFESSOR
DEPT. OF CSE
NEUB

Find output

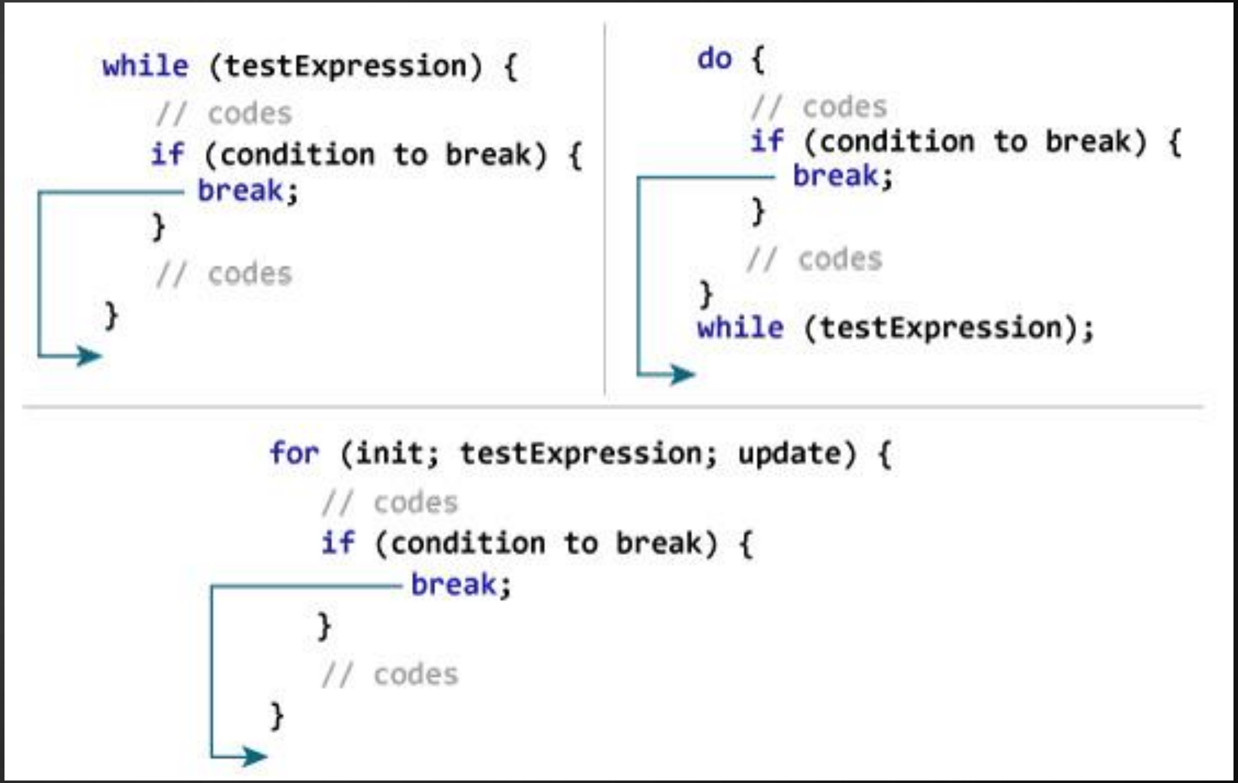
```
#include <stdio.h>
int main(){
    int i=0, x=0;
    for(i=0; i<10; i++)
    {
        if(i%2 == 1)
     x += i
        printf("%d ", x);
    }

    printf("\ni= %d, x= %d", i,x);

    return 0;
}
```

Find output

```
#include <stdio.h>

int main(){
    int i=10, x=0;
    while(i >= 0)
    {
        if(i%2 == 0)
          x += i
        printf("%d ", x);
        i--;
    }

    printf("\ni= %d, x= %d", i,x);

    return 0;
}
```

break statement

The break statement ends the loop immediately when it is encountered



```
while (testExpression) {
    // codes
    if (condition to break) {
        break;
    }
    // codes
}
```

```
do {
    // codes
    if (condition to break) {
        break;
    }
    // codes
}
while (testExpression);
```

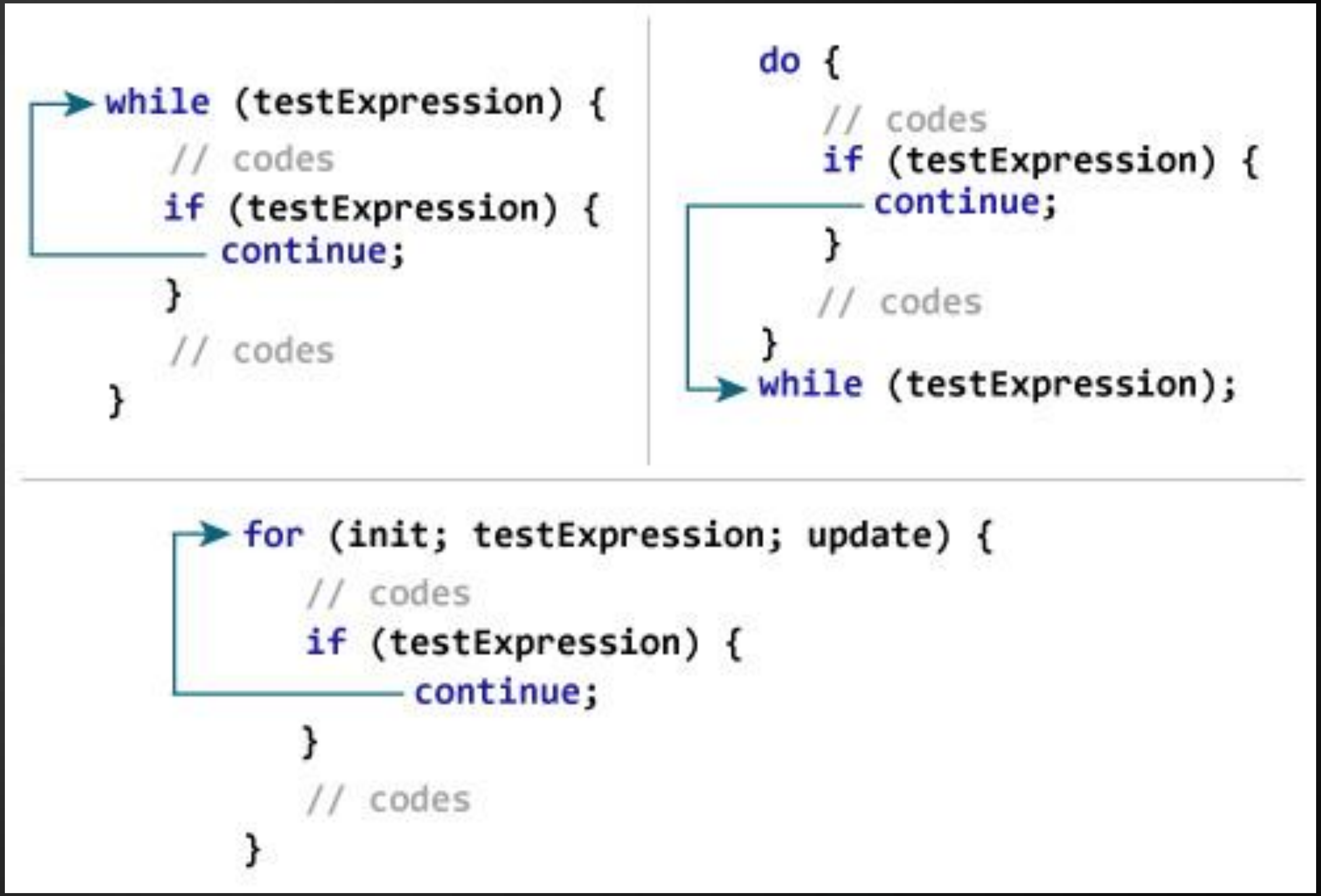
```
for (init; testExpression; update) {
    // codes
    if (condition to break) {
        break;
    }
    // codes
}
```

break statement

```
#include <stdio.h>
int main(){
   i = 1;

   While(i <= 10){
      if(i%5 == 0)
         break;
      printf(“%d”, i++);
   }

   printf(“The End”);
   return 0;
}
```

continue statement

The continue statement skips the current iteration of the loop and continues with the next iteration



```
while (testExpression) {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}
```

```
do {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}
while (testExpression);
```

```
for (init; testExpression; update) {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}
```

continue statement

```
#include <stdio.h>
int main(){
   i = 1;

   While(i <= 10){
      printf(“Before continue: %d\n”, i++);
      continue;
      printf(“After continue: %d\n”, i++);
   }

   printf(“The End”);
   return 0;
}
```

Switch…case

The switch statement allows to execute one code block among many alternatives

Syntax of switch...case

```
switch (expression)
{
    case constant1:
      // statements
      break;

    case constant2:
      // statements
      break;
    .
    .
    .
    default:
      // default statements
}
```