If-then-else vs Switch-case

There are two really noticeable differences here in comparison between if-then-else and switch. Firstly, break statement is used in switch to exit from case block and secondly there are no equality, inequality, greater than or less than comparisons to be seen in switch which are used in if-then-else.

break Statement

In Javascript programming, break is used in terminating the loop immediately after it is encountered. The break statement is used with conditional if statement.

Syntax of break statement

```
break;
```

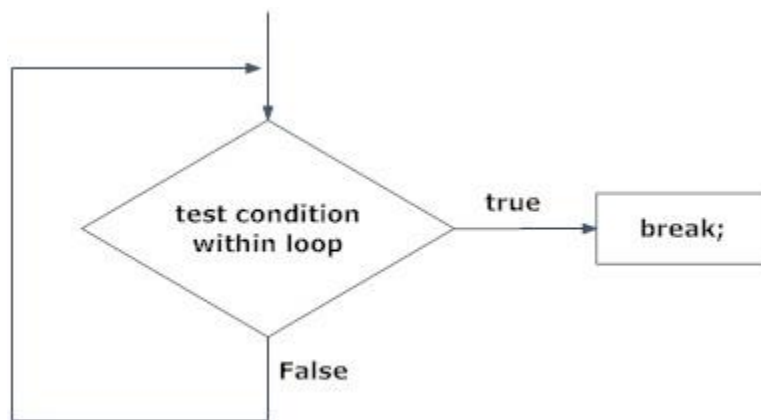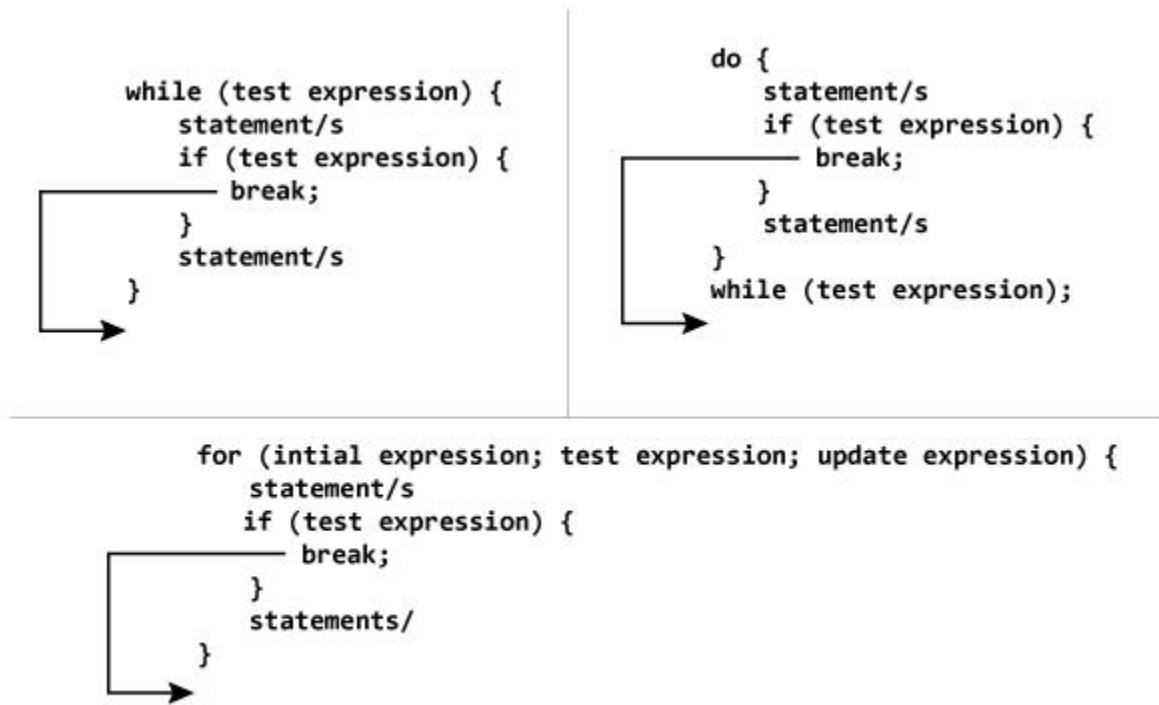The break statement can be used in terminating all three loops for, while and do...while loops.

Figure: Flowchart of break statement

The figure below explains the working of break statement in all three type of loops.

```
            while (test expression) {
                statement/s
                if (test expression) {
          ┌──────── break;
          │     }
          │     statement/s
          │  }
          └──▶
```

```
          do {
              statement/s
              if (test expression) {
        ┌──────── break;
        │     }
        │     statement/s
        │  }
        │  while (test expression);
        └──▶
```

```
            for (intial expression; test expression; update expression) {
                statement/s
                if (test expression) {
          ┌──────── break;
          │     }
          │     statements/
          │  }
          └──▶
```

NOTE: The break statment may also be used inside body of else statement.

continue Statement

It is sometimes desirable to skip some statements inside the loop. In such cases, continue statements are used.

Syntax of continue Statement

`continue;`

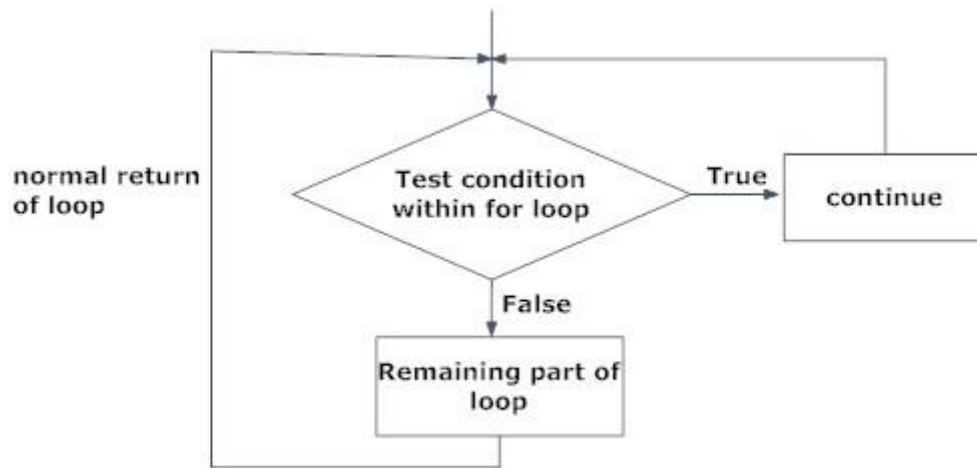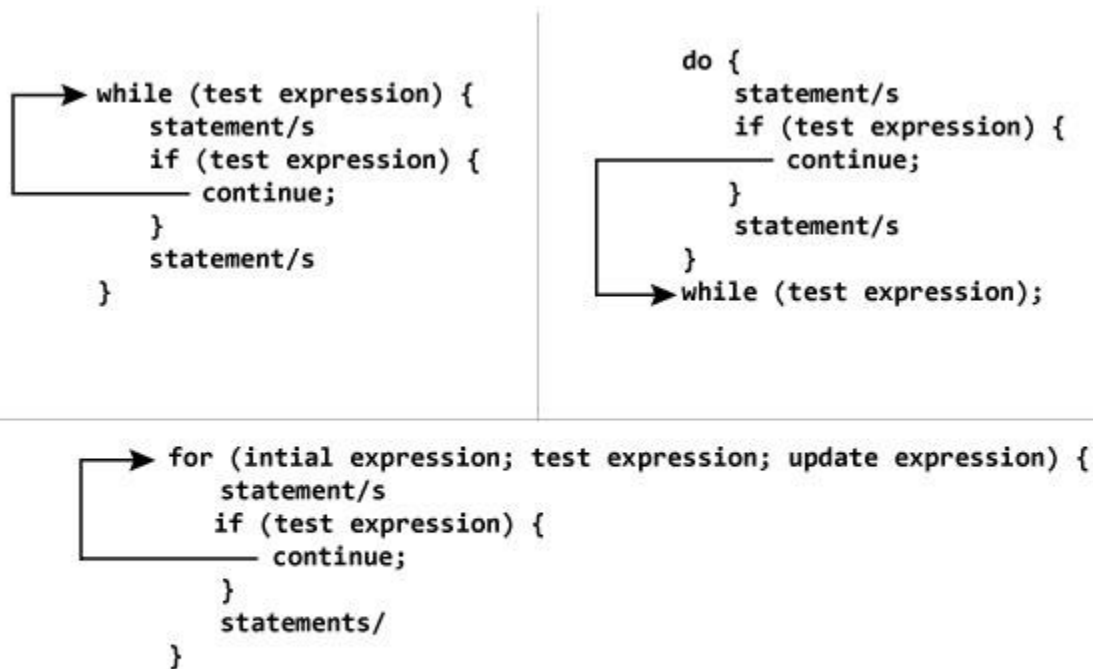Just like break, continue is also used with conditional if statement.

Fig: Flowchart of continue statement

For better understanding of how continue statements works in C programming. Analyze the figure below which bypasses some code/s inside loops using continue statement.

```
  ┌─► while (test expression) {
  │       statement/s
  │       if (test expression) {
  └─────── continue;
          }
          statement/s
      }
```

```
do {
        statement/s
        if (test expression) {
  ┌────── continue;
  │       }
  │       statement/s
  │   }
  └─►while (test expression);
```

```
  ┌─► for (intial expression; test expression; update expression) {
  │       statement/s
  │       if (test expression) {
  └─────── continue;
          }
          statements/
      }
```

NOTE: The continue statment may also be used inside body of else statement.