

① Which loop is guaranteed to execute at least once?

- A. for
- B. while.
- C. do-while.
- D. foreach

② How many times will this loop run?

```
for (int i=1; i<=5; i++) {}
```

- A. 4
- B. 5
- C. 6
- D. infinite

③ Which of these is an entry-controlled loop?

- A. while
- B. do-while.
- C. goto
- D. break.

④ What will the output be?

```
int i = 0;  
while (i<3) {  
    Print ("%d", i);  
    i++;  
}
```

- A. 012
- B. 123
- C. 001
- D. 013

- ⑤ which keyword stops the current iteration and jumps to the next?
- A. exit
 - B. break
 - C. stop
 - D. continue.
- ⑥ which loop is best when the number of iterations is known?
- A. while.
 - B. do-while.
 - C. for
 - D. nested loop.
- ⑦ what does the following code print?
- ```
for (int i=5; i>0; i--) {
 printf("%d", i);
}
```
- A. 1 2 3 4 5
  - B. 5 4 3 2 1
  - C. 5 4 2 1
  - D. error.
- ⑧ what type of loop is allowed inside another loop?
- A. only for inside for
  - B. only while inside for
  - C. only do-while inside while
  - D. All types (nested loops)

⑨ what is the output?

```
int i=1;
while (i<=5)
 i++;
Print ("%d", i);
```

A. 5

B. 6

C. loop never ends

D. 1

⑩ what will happen?

```
for (;;) {}.
```

A. Syntax error

B. executes 1 time after compilation with zero value

C. infinite loop

D. Depends on compiler.

⑪ which statement exits a loop completely?

A. break.

B. continue.

C. skip

D. switch.

⑫ output?

```
for (int i=1; i<5; i+=2)
 printf ("%d", i);
```

A. 1 2 3 4

B. 1 3 5

C. 1 3

D. 2 4

- ⑬ which loop executes at least one time?
- A. for
  - B. while
  - C. do-while.
  - D. none.

- ⑭ what is "nested loop"?
- A. Loop with multiple conditions.
  - B. Loop inside another loop
  - C. Loop without condition.
  - D. Loop outside a function.

- ⑮ output?

```
int sum=0;
for(int i=1; i<=3; i++)
 sum+=i;
printf ("%d", sum);
```

- A. 3
- B. 6
- C. 10
- D. 1

- ⑯ which loop is better for menu-driven programs?

- A. for
- B. while
- C. do-while.
- D. nested.

(17) What happens if loop condition is always true?

- A. Compilation error
- B. Program ends normally
- C. Infinite loop
- D. Loop runs twice.

(18) What does continue do?

- A. Ends loop
- C. Skips current iteration.
- C. Exits program.
- D. Restarts loop from beginning

(19) Identify the infinite loop.

- A. `for(i=0; i<10; i++)`
- B. `while (i>10)`
- C. `do{}while(1);`
- D. `for(i=1; i--)`

(20) What is the output?

```
int i=0;
do {
 printf ("%d", i);
 i++;
} while (i<2);
```

- A. 0
- B. 01
- C. 1 2
- D. No output