

DSA-3

Program: Print Even Numbers:-

Approach 1

```
i=2 4 6
for (int i = 2; i <= 10; i = i+2) {
    cout << i << " ";
}
```

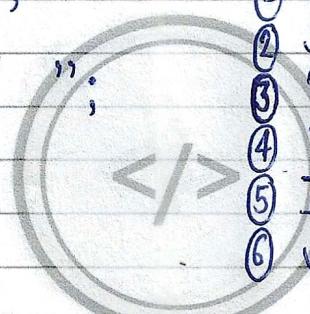
Approach 2

```
1%2 != 0
2%2 == 0
3%2 != 0
4%2 == 0
for (int i = 1; i <= 10; i++) {
    if (i%2 == 0)
        cout << i << " ";
```

Algorithm:-

- ① Start
- ② Initialize $i = 2$
- ③ Repeat loop while $i \leq 10$
- ④ Print i
- ⑤ Increment i by 2
- ⑥ Stop

Algorithm:-

- 
- ① Start
 - ② Initialize $i = 1$
 - ③ Repeat loop while $i \leq 10$
 - ④ if $i \% 2 == 0$, print i
 - ⑤ Increment i by 1
 - ⑥ Stop

Program: Print Alphabets from a to z:-

```
char i = 'a';
for (char i = 'a'; i <= 'z'; i++) {
    cout << i << " ";
}
```

\downarrow
gg
 \downarrow
b

Algorithm:-

- ① Start
- ② Initialize $i = 'a'$
- ③ Repeat loop while $i \leq 'z'$
- ④ Point i
- ⑤ Increment i by 1
- ⑥ Stop

Program: Sum of first 'n' natural numbers:-

```
int sum = 0;
for (int i = 1; i <= 10; i++) {
    sum += i;
}
```

\leftarrow
int=1, sum=1
 \leftarrow
int=2, sum=3
 \leftarrow
int=3, sum=6

Algorithm:-

- ① Start
- ② Initialize $i = 1$, $sum = 0$
- ③ Repeat loop while $i \leq 10$
- ④ Add i to sum
- ⑤ Increment i by 1
- ⑥ Print sum
- ⑦ Stop

Note

Why do we initialize $sum = 0$?

It contains a random value from memory.

Program: Sum of Squares of first 'n' Natural Numbers:-

```
int sum = 0;  
for (int i = 1; i <= 5; i++) {  
    sum += i * i;  
}  
cout << sum;  
i = 3  
sum = 14
```

Algorithm:-

- ① Start
- ② Initialize $i=1$, $sum=0$
- ③ Repeat loop while $i \leq 5$
- ④ Calculate $i \times i$ and add to sum
- ⑤ Increment i by 1
- ⑥ Print sum
- ⑦ Stop

Program: Reverse Counting :-

```
i = 10  
for (int i = 10; i >= 1; i--) {  
    cout << i << " ";  
}  
// 10
```

Algorithm:-

- ① Start
- ② Initialize $i=10$
- ③ Repeat loop while $i \geq 1$
- ④ Print i
- ⑤ Decrement i by 1
- ⑥ Stop

Program: Multiplication Table :-

```
5x1 - 5 int num = 5;  
5x2 - 10 for (int i = 1; i <= 10; i++) {  
5x3 - 15    cout << num << " x " << i << " = " << (num * i) << endl;  
5x4 = 20 }
```

→ Nested for Loop

```
for (initialization; condition; update) {  
    for (initialization; condition; update) {  
        // Code to repeat  
    }  
}
```

Pattern:- 1 2 3 4 5

$i=1$ 1 2 3 4 5

$j=1 2 3$ 1 2 3 4 5

4 5 6 1 2 3 4 5

$i=2$ 1 2 3 4 5

$j=1 2 3$ 1 2 3 4 5

4 5 6 1 2 3 4 5

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

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

```
        cout << j << " ";
```

```
    }
```

```
    cout << endl;
```

```
}
```

Pattern 2: a b c d e

```

int i=1
j=1
j=2
j=3
    
```

for($\text{int } i=1; i \leq 5; i++$) {
 for($\text{char } j='a'; j \leq 'e'; j++$) {
 cout << j << " ";
 }
 cout << endl;
}

Pattern 3: * * * * *

```

j=1
j=1
j=2
j=3
    
```

for($\text{int } i=1; i \leq 5; i++$) {
 for($\text{int } j=1; j \leq 5; j++$) {
 cout << "*" << ",";
 }
 cout << endl;
}

Pattern 4: *

```

j=1
j=1 to 1
j=2
j=1 to 2
    
```

for($\text{int } i=1; i \leq 5; i++$) {
 for($\text{int } j=1; j \leq i; j++$) {
 cout << "* " << ",";
 }
 cout << endl;
}

Pattern 5: * * * * *

```

j=5
j=1 to 5
j=4
j=1 to 4
    
```

for($\text{int } i=5; i \geq 1; i--$) {
 for($\text{int } j=1; j \leq i; j++$) {
 cout << "*" << ",";
 }
 cout << endl;
}

Pattern 6:

*	for($\text{int } i=1; i \leq 5; i++$) {
**	for($\text{int } j=1; j \leq 5-i; j++$) {
***	cout << "* "; }
****	for($\text{int } j=1; j \leq 2; j++$) {
*****	cout << "* "; }
	cout << endl; }

Yash Saini