

40-50 patterns

-star patterns

1> Square pattern

```
#include <stdio.h>
```

```
int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("* ");
        }
        printf("\n");
    }
}
```

2>empty square

```
#include <stdio.h>
```

```
int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if (i == 0 || i == n - 1 || j == 0 || j == n - 1) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
}
```

3>Right triangle

```
#include <stdio.h>
```

```
int main() {
    int n = 5;
```

```

    for (int i = 0; i < n; i++) {
        for (int j = 0; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
}

```

4>Inverted right angle triangle

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            printf("* ");
        }
        printf("\n");
    }
}

```

5>empty right angle triangle

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j <= i; j++) {
            if (j == 0 || i == n - 1 || j == i) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
}

```

6>reverse empty triangle

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            if (j == i || i == 0 || j == n - 1) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
}

```

7>Pyramid

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        for (int k = 0; k < (2 * i + 1); k++) {
            printf("*");
        }
        printf("\n");
    }
}

```

8>reverse pyramid

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < i; j++) {
            printf(" ");
        }
    }
}

```

```

        for (int k = i; k < (2 * (n - i) - 1); k++) {
            printf("*");
        }
        printf("\n");
    }
}

```

9>empty pyramid
#include <stdio.h>

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        for (int k = 0; k < (2 * i + 1); k++) {
            if (i == n - 1 || k == 0 || k == (2 * i)) {
                printf("*");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
}

```

10>empty reverse pyramid
#include <stdio.h>

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < i; j++) {
            printf(" ");
        }
        for (int k = i; k < (2 * (n - i) - 1); k++) {
            if (i == 0 || k == i || k == (2 * (n - i) - 2)) {
                printf("*");
            } else {

```

```

        printf(" ");
    }
}
printf("\n");
}
}

```

11>Diamond

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        for (int k = 0; k < (2 * i + 1); k++) {
            printf("*");
        }
        printf("\n");
    }
    for (int i = n - 2; i >= 0; i--) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        for (int k = 0; k < (2 * i + 1); k++) {
            printf("*");
        }
        printf("\n");
    }
}

```

12>empty diamond

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {

```

```

        printf(" ");
    }
    for (int k = 0; k < (2 * i + 1); k++) {
        if (i == n - 1 || k == 0 || k == (2 * i)) {
            printf("*");
        } else {
            printf(" ");
        }
    }
    printf("\n");
}
for (int i = n - 2; i >= 0; i--) {
    for (int j = i; j < n; j++) {
        printf(" ");
    }
    for (int k = 0; k < (2 * i + 1); k++) {
        if (i == 0 || k == 0 || k == (2 * i)) {
            printf("*");
        } else {
            printf(" ");
        }
    }
    printf("\n");
}
}

```

13>mirror right angle triangle

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        for (int k = 0; k <= i; k++) {
            printf("* ");
        }
        printf("\n");
    }
}

```

```
}  
}
```

-Number pattern

14>number triangle

```
#include <stdio.h>
```

```
int main() {  
    for (int i = 1; i <= 5; i++) {  
        for (int j = 1; j <= i; j++) {  
            printf("%d ", j);  
        }  
        printf("\n");  
    }  
    return 0;  
}
```

15>number square

```
#include <stdio.h>
```

```
int main() {  
    int N, i, j;  
  
    printf("Enter the number of rows and columns: ");  
    scanf("%d", &N);  
  
    for (i = 1; i <= N; i++) {  
        for (j = 1; j <= N; j++) {  
            printf("%d ", j);  
        }  
        printf("\n");  
    }  
  
    return 0;  
}
```

16>pyramid pattern

```
#include <stdio.h>
```

```
int main() {  
    int n = 5;
```

```

for (int i = 1; i <= n; i++) {
    for (int j = i; j < n; j++) {
        printf(" ");
    }
    for (int j = 1; j <= i; j++) {
        printf("%d ", j);
    }
    printf("\n");
}
return 0;
}

```

17>Pascal triangle

#include <stdio.h>

```

int main() {
    int n = 5;
    for (int line = 0; line < n; line++) {
        int C = 1; // Binomial coefficient
        for (int j = 0; j < line; j++) {
            printf(" ");
        }
        for (int j = 0; j <= line; j++) {
            printf("%d ", C);
            C = C * (line - j) / (j + 1);
        }
        printf("\n");
    }
    return 0;
}

```

18>Number diamond

#include <stdio.h>

```

int main() {
    int n = 5;
    for (int i = 1; i <= n; i++) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
    }
}

```



```

    }
    for (int j = 1; j <= (2 * i - 1); j++) {
        printf("%d", i);
    }
    printf("\n");
}
for (int i = n - 1; i >= 1; i--) {
    for (int j = n; j > i; j--) {
        printf(" ");
    }
    for (int j = 1; j <= (2 * i - 1); j++) {
        printf("%d", i);
    }
    printf("\n");
}
return 0;
}

```

19>right angle number triangle

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("%d ", i);
        }
        printf("\n");
    }
    return 0;
}

```

20>right angle triangle with 1 and 0

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("%d ", i % 2 == 0 ? 0 : 1);
        }
    }
}

```

```

    }
    printf("\n");
}
return 0;
}

```

21>decrease number triangle

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= i; j++) {
            printf("%d ", j);
        }
        printf("\n");
    }
    return 0;
}

```

22>pyramid of number

```
#include <stdio.h>
```

```

int main() {
    int n = 5; // Number of rows
    for (int i = 1; i <= n; i++) {
        // Print spaces
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        // Print numbers
        for (int j = 1; j <= i; j++) {
            printf("%d ", j);
        }
        printf("\n");
    }
    return 0;
}

```

23>empty number square

```
#include <stdio.h>
```

```

int main() {
    int n = 5; // Size of the square
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n || j == 1 || j == n) {
                printf("%d ", j);
            } else {
                printf(" "); // Print space for hollow part
            }
        }
        printf("\n");
    }
    return 0;
}

```

24>number pyramid patter increment one
#include <stdio.h>

```

int main() {
    int n = 5; // Number of rows
    for (int i = 1; i <= n; i++) {
        // Print spaces
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        // Print numbers
        for (int j = 1; j <= i; j++) {
            printf("%d ", i);
        }
        printf("\n");
    }
    return 0;
}

```

25>number patter that looks like hourglass
#include <stdio.h>

```

int main() {
    int n = 5; // Number of rows

```

```
// Upper half
for (int i = n; i >= 1; i--) {
    for (int j = n; j > i; j--) {
        printf(" ");
    }
    for (int j = 1; j <= i; j++) {
        printf("%d ", j);
    }
    printf("\n");
}
```

```
// Lower half
for (int i = 2; i <= n; i++) {
    for (int j = n; j > i; j--) {
        printf(" ");
    }
    for (int j = 1; j <= i; j++) {
        printf("%d ", j);
    }
    printf("\n");
}
```

```
return 0;
```

```
}
```

26>+1 triangle

```
#include <stdio.h>
```

```
int main() {
    int n = 5; // Number of rows
    int num = 1; // Starting number
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("%d ", num);
            num++;
        }
        printf("\n");
    }
    return 0;
```

```
}
```

26>empty pyramid

```
#include <stdio.h>
```

```
int main() {  
    int n = 5; // Number of rows  
    for (int i = 1; i <= n; i++) {  
        for (int j = i; j < n; j++) {  
            printf(" ");  
        }  
        for (int j = 1; j <= (2 * i - 1); j++) {  
            if (j == 1 || j == (2 * i - 1) || i == n) {  
                printf("%d", i);  
            } else {  
                printf(" ");  
            }  
        }  
        printf("\n");  
    }  
    return 0;  
}
```

-Alphabetic

27>right angle triangle

```
#include <stdio.h>
```

```
int main() {  
    int i, j;  
    char alphabet = 'A';  
    for (i = 1; i <= 5; i++) {  
        for (j = 1; j <= i; j++) {  
            printf("%c ", alphabet);  
            alphabet++;  
        }  
        printf("\n");  
        alphabet = 'A'; // Reset alphabet for next row  
    }  
    return 0;  
}
```

28>inverse right angle triangle

```
#include <stdio.h>
```

```
int main() {
    int i, j;
    char alphabet = 'A';
    for (i = 5; i >= 1; i--) {
        for (j = 1; j <= i; j++) {
            printf("%c ", alphabet);
            alphabet++;
        }
        printf("\n");
        alphabet = 'A'; // Reset alphabet for next row
    }
    return 0;
}
```

29>Pyramid

```
#include <stdio.h>
```

```
int main() {
    int i, j, k;
    char alphabet = 'A';
    for (i = 0; i < 5; i++) {
        for (j = 4; j > i; j--) {
            printf(" "); // Print spaces
        }
        for (k = 0; k <= i; k++) {
            printf("%c ", alphabet + k);
        }
        printf("\n");
    }
    return 0;
}
```

30>empty pyramid

```
#include <stdio.h>
```

```
int main() {
    int i, j, k;
```

```

char alphabet = 'A';
for (i = 0; i < 5; i++) {
    for (j = 4; j > i; j--) {
        printf(" "); // Print spaces
    }
    for (k = 0; k <= i; k++) {
        if (k == 0 || k == i || i == 4) {
            printf("%c ", alphabet + k);
        } else {
            printf(" "); // Print space for hollow part
        }
    }
    printf("\n");
}
return 0;
}

```

31>Diamond

```
#include <stdio.h>
```

```

int main() {
    int n = 5;
    // Upper half
    for (int i = 1; i <= n; i++) {
        for (int j = i; j < n; j++) {
            printf(" ");
        }
        for (int j = 1; j <= (2 * i - 1); j++) {
            printf("%c", 'A' + i - 1);
        }
        printf("\n");
    }
    // Lower half
    for (int i = n - 1; i >= 1; i--) {
        for (int j = n; j > i; j--) {
            printf(" ");
        }
        for (int j = 1; j <= (2 * i - 1); j++) {

```

```

        printf("%c", 'A' + i - 1);
    }
    printf("\n");
}
return 0;
}

```

32>Square

```
#include <stdio.h>
```

```

int main() {
    int n = 5; // Size of the square
    char alphabet = 'A';
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("%c ", alphabet);
        }
        printf("\n");
    }
    return 0;
}

```

33>revere alphabet

```
#include <stdio.h>
```

```

int main() {
    char alphabet;
    for (alphabet = 'Z'; alphabet >= 'A'; alphabet--) {
        printf("%c ", alphabet);
    }
    printf("\n");
    return 0;
}

```

34>