

Pattern Generator and Switch Case

26-10-25

Aim – Implement a program to print patterns and a program to display 4 digit number in words using switch case statement

Theory –

For Loop - The for loop executes a block of code repeatedly for a specified number of times, simplifying iteration and making repetitive tasks easier and faster.

Switch Case - The switch case allows multi-way branching based on variable values, making code cleaner, efficient, and easier to read than multiple if-else statements.

A1.

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

```
void main() {
```

```
    int n, i, j, space;
```

```
    printf("Enter number of rows for the upper half: ");
```

```
    scanf("%d", &n);
```

```
    for (i = 1; i <= n; i++) {
```

```
        for (space = i; space < n; space++){
```

```
            printf(" ");
```

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

```
            printf("* ");
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    printf("\n");
```

```
    for (i = n - 1; i >= 1; i--) {
```

```
        for (space = n; space > i; space--){
```

```
            printf(" ");
```

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

```
            printf("* ");
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
Enter number of rows for the upper half: 5
  *
 * *
* * *
* * * *
* * * * *

 * * * *
  * * *
   * *
    *

Process returned 0 (0x0)   execution time : 1.457 s
Press any key to continue.
}
```

A2.

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

```
void main() {
```

```
    int n, i, j, space;
```

```
    printf("Enter the number of rows for the upper half of diamond: ");
```

```
    scanf("%d", &n);
```

```
    for (i = 1; i <= n; i++) {
```

```
        for (space = i; space < n; space++){
```

```
            printf(" ");
```

```
        }
```

```
        for (j = 1; j <= (2 * i - 1); j++){
```

```
            printf("*");
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    for (i = n - 1; i >= 1; i--) {
```

```
        for (space = n; space > i; space--){
```

```
            printf(" ");
```

```
        }
```

```
        for (j = 1; j <= (2 * i - 1); j++) {
```

```
            printf("*");
```

```
        }
```

```
        printf("\n");
```

```
}  
}
```

```
Enter the number of rows for the upper half of diamond: 5  
  *  
 ***  
*****  
*****  
*****  
*****  
*****  
  ***  
  *  
  
Process returned 10 (0xA)   execution time : 1.229 s  
Press any key to continue.  
|
```

A3.

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

```
void main() {
```

```
    int num, digit1, digit2, digit3, digit4;
```

```
    printf("Enter a 4-digit number: ");
```

```
    scanf("%d", &num);
```

```
    if (num < 1000 || num > 9999) {
```

```
        printf("Please enter a valid 4-digit number.\n");
```

```
    } else {
```

```
        digit1 = num / 1000;
```

```
        digit2 = (num / 100) % 10;
```

```
        digit3 = (num / 10) % 10;
```

```
        digit4 = num % 10;
```

```
        printf("In words: ");
```

```
        switch (digit1) {
```

```
            case 1: printf("One"); break;
```

```
            case 2: printf("Two"); break;
```

```
case 3: printf("Three"); break;
case 4: printf("Four"); break;
case 5: printf("Five"); break;
case 6: printf("Six"); break;
case 7: printf("Seven"); break;
case 8: printf("Eight"); break;
case 9: printf("Nine"); break;
}
printf(" Thousand ");
```

```
switch (digit2) {
    case 0: break;
    case 1: printf("One"); break;
    case 2: printf("Two"); break;
    case 3: printf("Three"); break;
    case 4: printf("Four"); break;
    case 5: printf("Five"); break;
    case 6: printf("Six"); break;
    case 7: printf("Seven"); break;
    case 8: printf("Eight"); break;
    case 9: printf("Nine"); break;
}
```

```
printf(" Hundred ");
```

```
if (digit3 == 1) {
    switch (digit4) {
        case 0: printf("Ten"); break;
        case 1: printf("Eleven"); break;
        case 2: printf("Twelve"); break;
        case 3: printf("Thirteen"); break;
        case 4: printf("Fourteen"); break;
        case 5: printf("Fifteen"); break;
        case 6: printf("Sixteen"); break;
        case 7: printf("Seventeen"); break;
```

```
        case 8: printf("Eighteen"); break;
        case 9: printf("Nineteen"); break;
    }
} else {
    switch (digit3) {
        case 2: printf("Twenty "); break;
        case 3: printf("Thirty "); break;
        case 4: printf("Forty "); break;
        case 5: printf("Fifty "); break;
        case 6: printf("Sixty "); break;
        case 7: printf("Seventy "); break;
        case 8: printf("Eighty "); break;
        case 9: printf("Ninety "); break;
    }

    switch (digit4) {
        case 1: printf("One"); break;
        case 2: printf("Two"); break;
        case 3: printf("Three"); break;
        case 4: printf("Four"); break;
        case 5: printf("Five"); break;
        case 6: printf("Six"); break;
        case 7: printf("Seven"); break;
        case 8: printf("Eight"); break;
        case 9: printf("Nine"); break;
    }
}
}
printf("\n");
}
```

```
Enter a 4-digit number: 2371
In words: Two Thousand Three Hundred Seventy One

Process returned 10 (0xA)   execution time : 3.575 s
Press any key to continue.
|
```

Conclusion

In this assignment, I experimented with creating patterns using loops and converting four-digit numbers into words with the switch-case statement. It gave me a clearer understanding of how loops and decision-making work in C, boosting my programming confidence and problem-solving skills.