Assignment 01

```cpp
#include <bits/stdc++.h>
using namespace std;
void patternA()
{

    // * * * * *
    // *       *
    // *       *
    // *       *
    // *       *
    // * * * * *
    cout << "\n\nPattern A \n";
    for (int i = 1; i <= 6; i++)
    {
        for (int j = 1; j <= 5; j++)
        {
            if (i == 1 || j == 1 || i == 6 || j == 5)
            {
                cout << "* ";
            }
            else
            {
                cout << "  ";
            }
        }
        cout << endl;
    }
}
void patternB()
{
    // 1
    // 2 3
    // 4 5 6
    // 7 8 9 10
    cout << "\n\nPattern B \n";

    for (int i = 1; i < 5; i++)
    {
        for (int j = 0; j < i; j++)
        {
            cout << i + j << " ";
        }
        cout << endl;
    }
}
void patternC()
{
    //    1
    //   1 1
    //  1 2 1
```

```
    // 1 3 3 1
    cout << "\n\nPattern C \n";

    int rows = 4;
    int coef = 1;

    for (int i = 0; i < rows; i++)
    {
        for (int space = 1; space < rows - i; space++)
        {
            cout << " ";
        }
        coef = 1;
        for (int j = 0; j <= i; j++)
        {
            cout << coef << " ";
            coef = coef * (i - j) / (j + 1);
        }
        cout << endl;
    }
}
void patternD()
{
    // A
    // A B
    // A B C
    // A B C D
    // A B C D E
    cout << "\n\nPattern D \n";

    for (int i = 1; i < 6; i++)
    {
        char ch = 'A';
        for (int j = 1; j <= i; j++)
        {
            cout << ch++ << " ";
        }
        cout << endl;
    }
}
void patternE()
{
    //         *
    //       * * *
    //     * * * * *
    //   * * * * * * *
    // * * * * * * * * *
    cout << "\n\nPattern E \n";

    int rows = 5;

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

```
        for (int j = i; j < rows; j++)
        {
            cout << "  ";
        }
        for (int j = 1; j <= i; j++)
        {
            cout << "* ";
        }
        for (int j = 1; j < i; j++)
        {
            cout << "* ";
        }
        cout << endl;
    }
}
void patternF()
{
    //         1
    //       1 2 3
    //     1 2 3 4 5
    //   1 2 3 4 5 6 7
    // 1 2 3 4 5 6 7 8 9
    cout << "\n\nPattern F \n";

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

        for (int j = i; j < rows; j++)
        {
            cout << "  ";
        }
        for (int j = 1; j <= (2 * i - 1); j++)
        {
            cout << j << " ";
        }
        cout << endl;
    }
}
void patternG()
{
    //         A
    //       A B C
    //     A B C D E
    //   A B C D E F G
    // A B C D E F G H I
    cout << "\n\nPattern G \n";

    int rows = 5;
    for (int i = 1; i <= rows; i++)
    {
        char ch = 'A';
        for (int j = i; j < rows; j++)
```

```
        {
            cout << "  ";
        }
        for (int j = 1; j <= (2 * i - 1); j++)
        {
            cout << ch++ << " ";
        }
        cout << endl;
    }
}

int main()
{
    patternA();
    patternB();
    patternC();
    patternD();
    patternE();
    patternF();
    patternG();
    return 0;
}
```

Output: PS D:\Fullstack-Java-FirstBit-Solutions> & 'c:\Users\bhagv\.vscode\..\TDM-GCC-64\bin\gdb.exe' '--interpreter=mi'

Pattern A

* * * * *

* *

* *

* *

* *

* * * * *

Pattern B

1

2 3

3 4 5

4 5 6 7

Pattern C

  1

 1 1

1 2 1

1 3 3 1

Pattern D

A

A B

A B C

A B C D

A B C D E

Pattern E

```
    *
   * * *
  * * * * *
 * * * * * * *
* * * * * * * * *
```

Pattern F

```
    1
   1 2 3
  1 2 3 4 5
 1 2 3 4 5 6 7
1 2 3 4 5 6 7 8 9
```

Pattern G

```
    A
   A B C
  A B C D E
 A B C D E F G
A B C D E F G H I
```

PS D:\Fullstack-Java-FirstBit-Solutions>