Supreme-Batch-Debug-Exercise-C++ (Week-2)

NOTE: **The code snippet given may have compile time, runtime or logical errors.**

How to attempt Debugging Exercise?

1. Copy the code to your code editor (e.g. VS Code).
2. Add relevant header files like "#include <iostream>" etc.
3. Run the code.
4. You will notice the expected output is not printing at the console.
5. Apply your smart coder mind to Debug the code.
6. **Warning:** Only see the solution after you have tried enough.

1. **Add integers from 1 to N and display the sum on console.**

```
void main(){
    int n;cin>>n;
    int8_t sum=0;
    for(int i=0;i<n;++i){
        sum+=i;
    }
    cout<<sum<<endl;
    return 0;
}
```

```
void main(){
    int n;cin>>n;
    int8_t sum=0;
    for(int i=1;i<=n;++i){
        sum+=i;
    }
    cout<<(int)sum<<endl;
    return 0;
}
```

2. Print full pyramid like an Equilateral Triangle

```
#include <iostream>
using namespace std;
int main()
{
int k, n;
cout << "Enter the number of rows : ";
cin >> n;
cout << " ";
for (int i=1; i<=n; i++)
{

for (int j=1; j<=n-i; j++)
cout << " ';

for (j=1,k=i-1; j<=2*i-1; j++,k--)
{
if (1 || j <= k)
cout << j;
else
cout << k;
}
cout << endl;

cout << " ";

}
return 0;
}
```

```
#include <iostream>
using namespace std;
int main()
{
int k, n;
cout << "Enter the number of rows : ";
cin >> n;
cout << " ";
for (int i=1; i<=n; i++)
{

for (int j=1; j<=n-i; j++)
cout << " ";

for (j=1,k=2*i-1; j<=2*i-1; j++,k--)
{
if (j <= k)
cout << j;
else
cout << k;
}
cout << endl;

cout << " ";

}
return 0;
}
```

3. **Left Triangle star Pattern**

```
E.g., For N = 5
*
**
***
****
*****
```

```cpp
#include <iostream>
using namespace std;

int main() {
  // size of the triangle
  int size = N;
  // loop to print the pattern
  for (int i = 0; i < size; i++) {
    // print column
    for (int j = 0; j < i; j++) {
      cout << "**";
    }
    cout << "\n";
  }
  return 0;
}
```

```cpp
#include <iostream>
using namespace std;

int main() {
  // size of the triangle
  int N; cin>>N;
  int size = N;
  // loop to print the pattern
  for (int i = 0; i < size; i++) {
    // print column
    for (int j = 0; j <= i; j++) {
      cout << "*";
    }
    cout << "\n";
  }
  return 0;
}
```

4. Reverse Pyramid star pattern.

```
e.g., N=5
*********
 *******
  *****
   ***
    *
```

```
#include <iostream>
using namespace std;

int main() {
  // size of the pyramid
  int size; cin>>size;
  for (int i = 0; i < size; i++) {
    // print spaces
    for (int j = 0; j < i; j++) {
      cout << "  ";
    }
    // print stars
    for (k = 0; k < 2 * size - 1; k++) {
      cout << "*";
    }
    cout << "\n";
  }
  return 0;
}
```

```cpp
#include <iostream>
using namespace std;

int main() {
  // size of the pyramid
  int size; cin>>size;
  for (int i = 0; i < size; i++) {
    // print spaces
    for (int j = 0; j < i; j++) {
      cout << " ";
    }
    // print stars
    for (int k = 0; k < 2 * (size - i) - 1; k++) {
      cout << "*";
    }
    cout << "\n";
  }
  return 0;
}
```

5. Reverse Pyramid star pattern.

```
 e.g., for size=6;
 ***   ***
***** *****
**********
 ********
  ******
   *****
    ***
     *
```

C++

```cpp
#include <iostream>
using namespace std;

int main() {
  // heart star pattern
  int size;
  cin>>size;

  for (int i = size / 2; i < size; i += 2) {
    // print first spaces
    for (int j = 1; j < size - i; j += 2) {
      cout << " ";
    }
    // print first stars
    for (int j = 0; j < i + 1; j++) {
      cout << "*";
    }
    // print second spaces
    for (int j = 1; j < size - i + 1; j++) {
      cout << " ";
```

```
    for (int i = size / 2; i < size; i += 2) {
      // print first spaces
      for (int j = 1; j < size - i; j += 2) {
        cout << " ";
      }
      // print first stars
      for (int j = 0; j < i + 1; j++) {
        cout << "*";
      }
      // print second spaces
      for (int j = 1; j < size - i + 1; j++) {
        cout << " ";
      }
      // print second stars
      for (int j = 1; j < i + 1; j++) {
        cout << "*";
      }
      cout << "\n";
    }
    // lower part
    // inverted pyramid
    for (int i = size; i > 0; i++) {
      for (int j = 0; j < size - i; j++) {
        cout << " ";
      }
      for (int j = 1; j < i * 2; j++) {
        cout << "*";
      }
      cout << "\n";
    }
    return 0;
}
```

```cpp
#include <iostream>
using namespace std;

int main() {
  // heart star pattern
  int size;
  cin>>size;

  for (int i = size / 2; i < size; i += 2) {
    // print first spaces
    for (int j = 1; j < size - i; j += 2) {
      cout << " ";
    }
    // print first stars
    for (int j = 1; j < i + 1; j++) {
      cout << "*";
    }
    // print second spaces
    for (int j = 1; j < size - i + 1; j++) {
      cout << " ";
    }
    // print second stars
    for (int j = 1; j < i + 1; j++) {
      cout << "*";
    }
    cout << "\n";
  }
  // lower part
  // inverted pyramid
  for (int i = size; i > 0; i--) {
    for (int j = 0; j < size - i; j++) {
      cout << " ";
    }
    for (int j = 1; j < i * 2; j++) {
      cout << "*";
    }
    cout << "\n";
  }
  return 0;
}
```

6. Convert given Binary number to Decimal.

```cpp
int binaryToDecimal(int b){
    int ans;
    int c=0;
    while(b){
        ans=(b % 10) * (1 << c++);
        b/=10;
    }
    return ans;
}
```

```
int binaryToDecimal(int b){
    int ans=0;
    int c=0;
    while(b){
        ans=ans+(b % 10) * (1 << c++);
        b/=10;
    }
    return ans;
}
```

7. Simple Calculator.

```
#include <iostream>
using namespace std;

int main() {
    char oper;
    float num1, num2;
    cout << "Enter an operator (+, -, *, /): ";
    cin >> oper;
    cout << "Enter two numbers: " << endl;
    cin >> num1 >> num2;

    switch (oper) {
        case +:
            cout << num1 << " + " << num2 << " = " << num1 + num2;
        case -:
            cout << num1 << " - " << num2 << " = " << num1 - num2;
        case /:
            cout << num1 << " * " << num2 << " = " << num1 * num2;
        case *:
            cout << num1 << " / " << num2 << " = " << num1 / num2;
        default:
            // operator is doesn't match any case constant (+, -, *, /)
            cout << "Error! The operator is not correct";
            break;
    }

    return 0;
}
```

```
#include <iostream>
using namespace std;

int main() {
    char oper;
    float num1, num2;
    cout << "Enter an operator (+, -, *, /): ";
    cin >> oper;
    cout << "Enter two numbers: " << endl;
    cin >> num1 >> num2;

    switch (oper) {
        case '+':
            cout << num1 << " + " << num2 << " = " << num1 + num2;
            break;
        case '-':
            cout << num1 << " - " << num2 << " = " << num1 - num2;
            break;
        case '*':
            cout << num1 << " * " << num2 << " = " << num1 * num2;
            break;
        case '/':
            cout << num1 << " / " << num2 << " = " << num1 / num2;
            break;
        default:
            // operator is doesn't match any case constant (+, -, *, /)
            cout << "Error! The operator is not correct";
            break;
    }

    return 0;
}
```