Quiz 4 (DS-A)
B

Roll # _____ **Name** _____

```
void evenFibonacciSum(int num) {
   int a = 0, b = 1, sum = 0;
   while (b <= num) {
      if (b % 2 == 0) {
         sum += b;}
      int next = a + b;
      a = b;
      b = next;}
   cout << "Sum of even Fibonacci numbers up to " << num << ": " <<
sum << endl;}
int main() {
   int totalSum = 0;
   for (int i = 1; i <= 5; ++i) {
      totalSum += i;
      cout << "Current totalSum: " << totalSum << endl;
      for (int j = 1; j <= 3; ++j) {
         if (j == 2) {
            evenFibonacciSum(i * j);}}}
   return 0;}
```

Output/Error :

```
Current totalSum: 1
Sum of even Fibonacci numbers up
to 2: 2
Current totalSum: 3
Sum of even Fibonacci numbers up
to 4: 2
Current totalSum: 6
Sum of even Fibonacci numbers up
to 6: 2
Current totalSum: 10
Sum of even Fibonacci numbers up
to 8: 10
Current totalSum: 15
Sum of even Fibonacci numbers up
to 10: 10
```

```
int main() {
   for (char i = 'A'; i <= 'A'; i++) {
      for (char j = 'A'; j <= 'A'; j++) {
         for (char k = 'A'; k <= 'A'; k++) {
            for (char l = 'A'; l <= 'B'; l++) {
               for (char m = 'A'; m <= 'B'; m++) {
                  for (char n = 'A'; n <= 'A'; n++) {
                     cout << i << j << k << l << m << n << endl;
}}}}}}
   return 0;}
```

Output/Error :

```
AAAAAA
AAAABA
AAABAA
AAABBA
```

```
void generateCombinations(string current, int depth, char start, char end) {
    if (depth == 3) {
        if (current[0] == current[1] && current[1] == current[2]) {
            cout << current << "#" << endl;
        } else {
            cout << current << "*" << endl;}
        return;}
    for (char c = start; c <= end; c++) {
        generateCombinations(current + c, depth + 1, start, end);}}
void printCombinations() {
    generateCombinations("", 0, 'A', 'B');}
int main() {
    printCombinations();
    return 0;}
```

Output/Error :

AAA#
AAB*
ABA*
ABB*
BAA*
BAB*
BBA*
BBB#

```
int main() {
    int n = 5;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n || j == 1 || j == n) {
                cout << "# ";
            } else if ((i + j) % 2 == 0) {
                if (i % 2 == 0) {
                    cout << "* ";
                } else {
                    cout << "@ ";}
            } else {
                cout << "- ";}}
        cout << endl;}
    for (int k = 0; k < 5; k++) {
        cout << "PF ";}
    cout << endl;
    return 0;}
```

Output/Error :

# # # # #
* - *
- @ -
* - *
# # # # #
PF PF PF PF PF

```
int main() {
    int n = 5;
    for (int i = 1; i <= n; i++) {
        for (int j = n; j > i; j--)
            cout << " ";
        for (int j = 1; j <= (2 * i - 1); j++) {
            if (j == 1 || j == (2 * i - 1)) {
                int sum = i + j;
                if (sum % 3 == 0)
                    cout << '*';
                else if (sum % 3 == 1)
                    cout << '#';
                else
                    cout << '@';
            } else {
                cout << ' ';}}
        cout << endl;}
    for (int i = n - 1; i >= 1; i--) {
        for (int j = n; j > i; j--)
            cout << " ";
        for (int j = 1; j <= (2 * i - 1); j++) {
            if (j == 1 || j == (2 * i - 1)) {
                int sum = i + j;
                if (sum % 3 == 0)
```

Output/Error :

```
                cout << '*';
            else if (sum % 3 == 1)
                cout << '#';
            else
                cout << '@';
        } else {
            cout << ' ';}}
    cout << endl;}
return 0;}
```

```
int main() {
    int n = 5;
    for (int i = n; i >= 0; i--) {
        for (int j = n; j > i; j--)
            cout << " ";
        int C = 1;
        for (int j = 0; j <= i; j++) {
            cout << C << " ";
            C = C * (i - j) / (j + 1);}
        cout << endl;}
    return 0;}
```

Output/Error

```
1 5 10 10 5 1
 1 4 6 4 1
  1 3 3 1
   1 2 1
    1 1
     1
```