General I/O Template

```c
#include <stdio.h>
int main() {
  int var;

  while (scanf("%d", &var) == 1) {
    // Process input
  }

  return 0;
}
```

Non-Tail Recursive Example

The following implementation is not tail recursive because the multiplication operation occurs after the recursive call returns.

```c
int factorial(unsigned int n) {
  if (n == 0) // Base case
    return 1;
  else
    return n * factorial(n - 1);
}
```

Tail Recursive Example (Accumulator)

The following implementation uses an accumulator to achieve tail recursion.

```c
int fact_tail(unsigned int n, unsigned int acc) {
  if (n == 0)
    return acc;
  return fact_tail(n - 1, acc * n);
}
int factorial(unsigned int n) {
  return fact_tail(n, 1);
}
```

Recursion With Arrays

The following implementation calculates the number of ways to make change for a given amount using recursion with arrays.

```c
int countWaysCoins(int coins[], int m, int amount) {
  if (amount == 0) {
    return 1;
  }

  if (amount < 0 || m <= 0) {
    return 0;
  }

  return countWaysCoins(coins, m - 1, amount) +            // Exclude the last coin
         countWaysCoins(coins, m, amount - coins[m - 1]);  // Use the last coin
}
```