

# TypeQuest

## Cognitive Function Score calculation logic - abstract:

```
int Ni, Ne, Fi, Fe, Si, Se = 0;
for each row/question:
    //what function was the question trying to determine (eg. Ni)
    get Question Target Function
    //what was the value of the button clicked?
    get Position of radiobutton or associated value
    //perform calculation
    increment Target Function Int by associated value of selected
radio button
```

## MBTI type calculation logic - abstract:

MBTI Overview

- **Type**: [MBTI Type]
- **Cognitive Function Stack**: [Dominant, Auxiliary, Tertiary, Inferior]
- **Description**: [Brief description of MBTI Type]

// Step 1: Calculate Cognitive Function Scores

```
int Ni, Ne, Fi, Fe, Si, Se, Ti, Te = 0;
```

```
for each row/question:
```

```
    get Question Target Function
```

```
    get Position of radiobutton or associated value
```

```
    increment Target Function Int by associated value of selected
radio button
```

// Step 2: Determine Dominant Functions and Alternate

```
functionScores = [(Ni, "Ni"), (Ne, "Ne"), (Fi, "Fi"), (Fe, "Fe"),
(Si, "Si"), (Se, "Se"), (Ti, "Ti"), (Te, "Te")]
```

```
sortedFunctions = sort functionScores by value in descending order
```

```
introvertedFunctions = []
```

```
extrovertedFunctions = []
```

```
for each function in sortedFunctions:
```

```
    if function is introverted:
```

```
        add function to introvertedFunctions
```

```
    else:
```

```
        add function to extrovertedFunctions
```

```
cognitiveFunctionStack = []
```

```
while cognitiveFunctionStack has less than 4 functions:
```

```
    if cognitiveFunctionStack has less than 4 functions and
extrovertedFunctions is not empty:
```

```

        add first element of extrovertedFunctions to
cognitiveFunctionStack
        if cognitiveFunctionStack has less than 4 functions and
introvertedFunctions is not empty:
            add first element of introvertedFunctions to
cognitiveFunctionStack

return cognitiveFunctionStack

```

## Enneagram type calculation logic - abstract:

```

### Enneagram Overview
- **Type**: [Enneagram Type]
- **Wing**: [Wing Type]
- **Description**: [Give a brief description of Enneagram Type and
Wing]

### Enneagram Type Calculation
int Type1, Type2, Type3, Type4, Type5, Type6, Type7, Type8, Type9 =
0;

for each row/question:
    // what type was the question trying to determine (e.g., Type 1)
    get Question Target Type
    // what was the value of the button clicked?
    get Position of radiobutton or associated value
    // perform calculation
    increment Target Type Int by associated value of selected radio
button

### Determining Dominant Enneagram Type
functionScores = [(Type1, "Type1"), (Type2, "Type2"), (Type3,
"Type3"), (Type4, "Type4"), (Type5, "Type5"), (Type6, "Type6"),
(Type7, "Type7"), (Type8, "Type8"), (Type9, "Type9")]

sortedFunctions = sort functionScores by value in descending order

dominantType = sortedFunctions[0]
wingType = sortedFunctions[1] // Typically the wing is one of the
adjacent types

// Enneagram Types Description
switch dominantType:
    case "Type1":
        description = "Type 1: The Reformer - principled,
purposeful, self-controlled, and perfectionistic."
    case "Type2":
        description = "Type 2: The Helper - generous, demonstrative,
people-pleasing, and possessive."
    // add descriptions for other types

```

```
return (dominantType, wingType, description)
```