Simple Algorithms in C++

Your Name

August 24, 2024

1 Introduction

This document presents several simple algorithms implemented in C++ using only basic control structures like loops and conditional statements. These algorithms avoid the use of functions, classes, objects, vectors, or complex functions.

2 Algorithms

1. Convert a Number from Base 10 to Base n

```cpp
#include <iostream>

int number = 156;
int base = 2;
int result[32];
int index = 0;

do {
    result[index] = number % base;
    number = number / base;
    index++;
} while (number > 0);

for (int i = index - 1; i >= 0; i--) {
    std::cout << result[i];
}
```

2. Convert a Number from Base n to Base 10

```cpp
#include <iostream>
#include <cmath>

int base = 2;
int number[] = {1, 0, 1, 0}; // Binary representation
int size = 4;
int result = 0;

for (int i = 0; i < size; i++) {
    result = result + number[i] * pow(base, size - i - 1);
}

```

```cpp
std::cout << result;
```

3. Swap Two Numbers

```cpp
#include <iostream>

int a = 5;
int b = 10;
int temp;

temp = a;
a = b;
b = temp;

std::cout << "a = " << a << ", b = " << b;
```

4. Factorize an Integer

```cpp
#include <iostream>

int number = 28;

for (int i = 1; i <= number; i++) {
    if (number % i == 0) {
        std::cout << i << " ";
    }
}
```

5. Convert an Integer to Roman Numerals

```cpp
#include <iostream>

int num = 1987;
while (num > 0) {
    if (num >= 1000) {
        std::cout << "M";
        num -= 1000;
    } else if (num >= 900) {
        std::cout << "CM";
        num -= 900;
    } else if (num >= 500) {
        std::cout << "D";
        num -= 500;
    } else if (num >= 400) {
        std::cout << "CD";
        num -= 400;
    } else if (num >= 100) {
        std::cout << "C";
        num -= 100;
    } else if (num >= 90) {
        std::cout << "XC";
        num -= 90;
    } else if (num >= 50) {
        std::cout << "L";
        num -= 50;
    } else if (num >= 40) {
        std::cout << "XL";
```

```
        num -= 40;
    } else if (num >= 10) {
        std::cout << "X";
        num -= 10;
    } else if (num >= 9) {
        std::cout << "IX";
        num -= 9;
    } else if (num >= 5) {
        std::cout << "V";
        num -= 5;
    } else if (num >= 4) {
        std::cout << "IV";
        num -= 4;
    } else {
        std::cout << "I";
        num -= 1;
    }
}
```

6. Compare Two Numbers

```
#include <iostream>

int a = 15;
int b = 10;

if (a > b) {
    std::cout << "a is greater than b";
} else if (a < b) {
    std::cout << "a is less than b";
} else {
    std::cout << "a is equal to b";
}
```

7. Sort a List of Numbers (Ascending/Descending)

```
#include <iostream>

int numbers[] = {5, 3, 8, 1, 2};
int size = 5;
bool ascending = true;

for (int i = 0; i < size - 1; i++) {
    for (int j = 0; j < size - i - 1; j++) {
        if (ascending && numbers[j] > numbers[j + 1]) {
            int temp = numbers[j];
            numbers[j] = numbers[j + 1];
            numbers[j + 1] = temp;
        } else if (!ascending && numbers[j] < numbers[j + 1])
    {
            int temp = numbers[j];
            numbers[j] = numbers[j + 1];
            numbers[j + 1] = temp;
        }
    }
}

```

```cpp
for (int i = 0; i < size; i++) {
    std::cout << numbers[i] << " ";
}
```

8. Calculate Happy Numbers

```cpp
#include <iostream>

int number = 19;
int temp, digit, sum;

do {
    sum = 0;
    while (number > 0) {
        digit = number % 10;
        sum += digit * digit;
        number /= 10;
    }
    number = sum;
} while (sum != 1 && sum != 4);

if (sum == 1) {
    std::cout << "Happy Number";
} else {
    std::cout << "Not a Happy Number";
}
```

9. Fibonacci Series

```cpp
#include <iostream>

int n = 10;
int a = 0, b = 1, c;
std::cout << a << " " << b << " ";
for (int i = 2; i < n; i++) {
    c = a + b;
    std::cout << c << " ";
    a = b;
    b = c;
}
```

10. Check if a Number is Even or Odd

```cpp
#include <iostream>

int n = 5;
if (n % 2 == 0) {
    std::cout << n << " is even." << std::endl;
} else {
    std::cout << n << " is odd." << std::endl;
}
```

11. Count Specific Digits in a Number

```cpp
#include <iostream>

```

```cpp
int number = 1234564564444678;
int digit = 4;
int count = 0;
while (number > 0) {
    if (number % 10 == digit) {
        count++;
    }
    number /= 10;
}
std::cout << "The digit " << digit << " appears " << count <<
    " times." << std::endl;
```

12. Calculate the Factorial of a Number

```cpp
#include <iostream>

int n = 5;
int factorial = 1;
for (int i = 1; i <= n; i++) {
    factorial *= i;
}
std::cout << "Factorial of " << n << " is " << factorial <<
    std::endl;
```

13. Sum Two Numbers

```cpp
#include <iostream>

int a = 5;
int b = 10;
int sum = a + b;
std::cout << "Sum: " << sum << std::endl;
```

14. Find the Greatest of Three Numbers

```cpp
#include <iostream>

int a = 10, b = 15, c = 5;
int greatest;
if (a > b) {
    if (a > c) {
        greatest = a;
    } else {
        greatest = c;
    }
} else {
    if (b > c) {
        greatest = b;
    } else {
        greatest = c;
    }
}
std::cout << "The greatest number is " << greatest << std::
    endl;
```

15. Sum of First n Natural Numbers

```cpp
#include <iostream>

int n = 10;
int sum = 0;
for (int i = 1; i <= n; i++) {
    sum += i;
}
std::cout << "Sum of first " << n << " natural numbers is " <<
     sum << std::endl;
```

16. Reverse a Number

```cpp
#include <iostream>

int n = 12345;
int reversed = 0;
while (n > 0) {
    reversed = reversed * 10 + n % 10;
    n /= 10;
}
std::cout << "Reversed number: " << reversed << std::endl;
```

17. Find the Greatest Common Divisor (GCD)

```cpp
#include <iostream>

int a = 48, b = 18;
while (a != b) {
    if (a > b) {
        a -= b;
    } else {
        b -= a;
    }
}
std::cout << "GCD is " << a << std::endl;
```

18. Check if a Number is a Palindrome

```cpp
#include <iostream>

int n = 12321;
int original = n;
int reversed = 0;
while (n > 0) {
    reversed = reversed * 10 + n % 10;
    n /= 10;
}
if (original == reversed) {
    std::cout << "The number is a palindrome." << std::endl;
} else {
    std::cout << "The number is not a palindrome." << std::
    endl;
}
```

19. Calculate the Power of a Number

```cpp
#include <iostream>

int base = 2, exp = 5;
int result = 1;
for (int i = 0; i < exp; i++) {
    result *= base;
}
std::cout << base << " raised to the power " << exp << " is "
    << result << std::endl;
```

20. Find the Largest Element in an Array

```cpp
#include <iostream>

int arr[] = {1, 5, 3, 9, 2};
int n = sizeof(arr) / sizeof(arr[0]);
int max = arr[0];
for (int i = 1; i < n; i++) {
    if (arr[i] > max) {
        max = arr[i];
    }
}
std::cout << "The largest element is " << max << std::endl;
```

21. Find the Smallest Element in an Array

```cpp
#include <iostream>

int arr[] = {1, 5, 3, 9, 2};
int n = sizeof(arr) / sizeof(arr[0]);
int min = arr[0];
for (int i = 1; i < n; i++) {
    if (arr[i] < min) {
        min = arr[i];
    }
}
std::cout << "The smallest element is " << min << std::endl;
```

22. Sum All Elements in an Array

```cpp
#include <iostream>

int arr[] = {1, 5, 3, 9, 2};
int n = sizeof(arr) / sizeof(arr[0]);
int sum = 0;
for (int i = 0; i < n; i++) {
    sum += arr[i];
}
std::cout << "The sum of all elements is " << sum << std::endl
    ;
```

23. Count the Number of Vowels in a String

```cpp
#include <iostream>
#include <string>

```

```
std::string str = "Hello World";
int count = 0;
for (int i = 0; i < str.length(); i++) {
    char c = tolower(str[i]);
    if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u') {
        count++;
    }
}
std::cout << "The number of vowels is " << count << std::endl;
```

24. Sort an Array (Descending Order - Method 1)

```
#include <iostream>

int arr[] = {1, 5, 3, 9, 2};
int n = sizeof(arr) / sizeof(arr[0]);

for (int i = 0; i < n-1; i++) {
    for (int j = 0; j < n-i-1; j++) {
        if (arr[j] < arr[j+1]) {
            int temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;
        }
    }
}
std::cout << "Sorted array in descending order: ";
for (int i = 0; i < n; i++) {
    std::cout << arr[i] << " ";
}
```

25. Sort an Array (Descending Order - Method 2)

```
#include <iostream>

int arr[] = {1, 5, 3, 9, 2};
int n = sizeof(arr) / sizeof(arr[0]);

for (int i = 0; i < n; i++) {
    int max_idx = i;
    for (int j = i+1; j < n; j++) {
        if (arr[j] > arr[max_idx]) {
            max_idx = j;
        }
    }
    int temp = arr[i];
    arr[i] = arr[max_idx];
    arr[max_idx] = temp;
}
std::cout << "Sorted array in descending order: ";
for (int i = 0; i < n; i++) {
    std::cout << arr[i] << " ";
}
```

26. Sort an Array (Descending Order - Method 3)

```cpp
#include <iostream>

int arr[] = {1, 5, 3, 9, 2};
int n = sizeof(arr) / sizeof(arr[0]);

for (int i = 0; i < n; i++) {
    for (int j = i+1; j < n; j++) {
        if (arr[i] < arr[j]) {
            int temp = arr[i];
            arr[i] = arr[j];
            arr[j] = temp;
        }
    }
}
std::cout << "Sorted array in descending order: ";
for (int i = 0; i < n; i++) {
    std::cout << arr[i] << " ";
}
```

27. Reverse a String

```cpp
#include <iostream>
#include <string>

std::string str = "Hello World";
std::string reversed = "";

for (int i = str.length() - 1; i >= 0; i--) {
    reversed += str[i];
}

std::cout << "Reversed string: " << reversed << std::endl;
```

28. Calculate the Sum of Digits of a Number

```cpp
#include <iostream>

int n = 12345;
int sum = 0;

while (n > 0) {
    sum += n % 10;
    n /= 10;
}

std::cout << "Sum of digits: " << sum << std::endl;
```

29. Check if a Year is a Leap Year

```cpp
#include <iostream>

int year = 2024;

if (year % 4 == 0) {
    if (year % 100 == 0) {
        if (year % 400 == 0) {
```

```
8               std::cout << year << " is a leap year." << std::
    endl;
9           } else {
10              std::cout << year << " is not a leap year." << std
    ::endl;
11          }
12      } else {
13          std::cout << year << " is a leap year." << std::endl;
14      }
15  } else {
16      std::cout << year << " is not a leap year." << std::endl;
17  }
```