

C++応用演習 2

第 1 回レポート課題

提 出 日 2018 年 11 月 27 日

学 籍 番 号 H T 1 7 A 1 0 4
氏 名 山 田 裕 輝
メールアドレス ht17a104@oecu.jp

1. 作成したプログラムのソースコード

```
1. #include <iostream>
2. #include <string>
3. #define GP2_PRIME_UPPER_LIMIT 1000000 //探索する値の上限値。
4. std::iostream;
5. std::string;
6.
7. //素数判定
8. bool IsPrime(int num)
9. {
10.     if (num < 2) return false;
11.     else if (num == 2) return true;
12.     else if (num % 2 == 0) return false; // 偶数はあらかじめ除く
13.
14.     double sqrtNum = sqrt(num);
15.     for (int i = 3; i <= sqrtNum; i += 2)
16.     {
17.         if (num % i == 0)
18.         {
19.             // 素数ではない
20.             return false;
21.         }
22.     }
23.     // 素数である
24.     return true;
25. }
26.
27. void tagai(unsigned int a, unsigned int d) {
28.     if (IsPrime(a) == false) {
```

```

29.         std::cout << "互いに素ではない";
30.     }
31.     else if (IsPrime(d) == false) {
32.         std::cout << "互いに素ではない";
33.     } else {
34.         std::cout << "互いに素";
35.     }
36. }
37.
38. /** 与えられた正整数 a と d と n に対して、この等差数列に含まれる n 番目の
39.  * 素数を返す。
40. */
41. int nth_prime(unsigned int a, unsigned int d, unsigned int n) {
42.     int sum = a;
43.     int c = 0;
44.     for (; c < n && sum < GP2_PRIME_UPPER_LIMIT;) {
45.         if (IsPrime(sum) == true) {
46.             c += 1;
47.             if (c == n) {
48.                 return sum;
49.             }
50.         }
51.         sum += d;
52.     }
53.     return sum;
54. }
55.
56. int main() {
57.     std::cout << nth_prime(367, 186, 151) << " : "; tagai(367, 186); std::cout << std::endl;
58.     std::cout << nth_prime(179, 10, 203) << " : "; tagai(179, 10); std::cout << std::endl;

```

```

59.     std::cout << nth_prime(271, 37, 39) << " : "; tagai(271, 37); std::cout << std::endl;
60.     std::cout << nth_prime(103, 230, 1) << " : "; tagai(103, 230); std::cout << std::endl;
61.     std::cout << nth_prime(27, 104, 185) << " : "; tagai(27, 104); std::cout << std::endl;
62.     std::cout << nth_prime(253, 50, 85) << " : "; tagai(253, 50); std::cout << std::endl;
63.     std::cout << nth_prime(1, 1, 1) << " : "; tagai(1, 1); std::cout << std::endl;
64.     std::cout << nth_prime(9075, 337, 210) << " : "; tagai(9075, 337); std::cout << std::endl;
65.     std::cout << nth_prime(307, 24, 79) << " : "; tagai(307, 24); std::cout << std::endl;
66.     std::cout << nth_prime(331, 221, 177) << " : "; tagai(331, 221); std::cout << std::endl;
67.     std::cout << nth_prime(259, 170, 40) << " : "; tagai(259, 170); std::cout << std::endl;
68.     std::cout << nth_prime(269, 58, 102) << " : "; tagai(269, 58); std::cout << std::endl;
69.
70.     return 0;
71. }

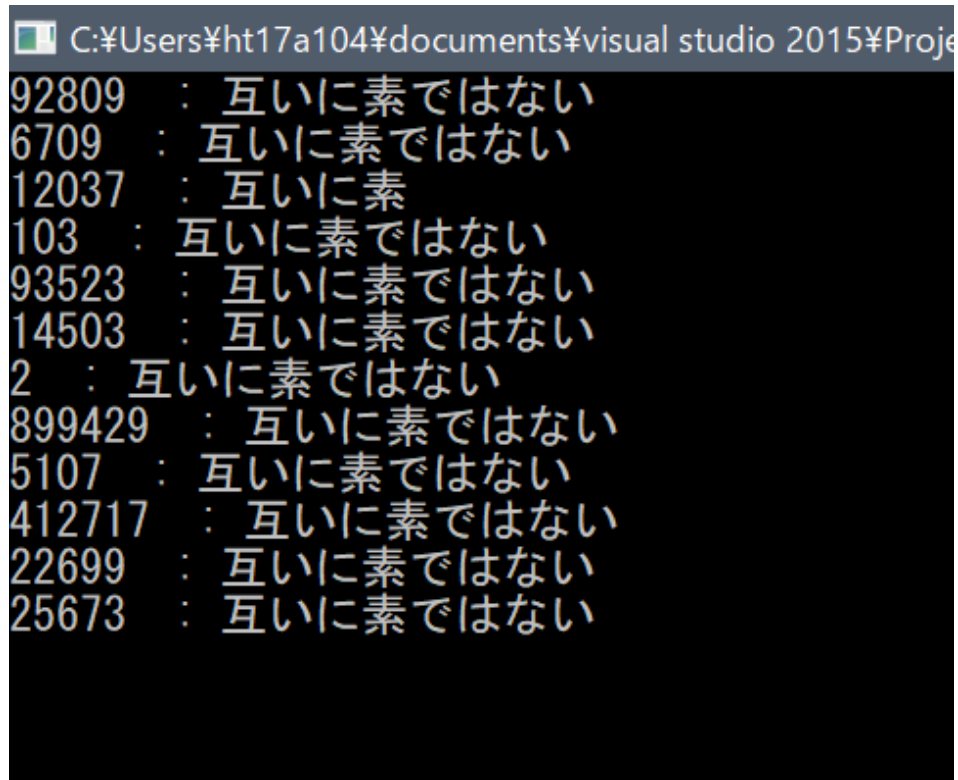
```

2. ソースコードの解説

- 与えられた正整数 **a** と **d** と **n** に対して、この等差数列に含まれる **n** 番目の素数を求めるプログラムを作った
- **bool** 型関数 **IsPrime** は与えられた整数型の引数が素数であるか判定する。
引数が素数であれば **true**、そうでなければ **false** を返す。
- **void** 型関数 **tagai** は整数型の 2 つの引数が互いに素であるかを判定する。
互いに素であれば **true**、そうでなければ **false** を返す。
- **int** 型関数 **nth_prime** は与えられた正整数 **a** と **d** と **n** に対して、**a+d** の等差数列に含まれる **n** 番目の素数を返す。
for 文では **n** 番目の素数を見つけるか、**a+d** が上限にいくまで繰り返すようになっている。
- **main()** には求める整数が入れられた **nth_prime** と **tagai** を呼び出している。

3. 実行結果

実行結果は以下のようになった。



```
C:\Users\ht17a104\documents\visual studio 2015\Proje
92809 : 互いに素ではない
6709 : 互いに素ではない
12037 : 互いに素
103 : 互いに素ではない
93523 : 互いに素ではない
14503 : 互いに素ではない
2 : 互いに素ではない
899429 : 互いに素ではない
5107 : 互いに素ではない
412717 : 互いに素ではない
22699 : 互いに素ではない
25673 : 互いに素ではない
```

4. 考察

入力した整数から正しい数値が返ってきたので正しくプログラムが書けたのではないかと思います。入力された a と d が互いに素であるか判定する際に、`nth_prime` と同じ一連の流れに入れようとした。しかし文字で出力することができなかったため、やむを得ず関数を分けることになったので、もう少し改善の余地があるのではないかと思います。

5. 感想

今回 `count` を使っていなかったが使うことで、さらに `nth_prime` を完結に書けたと思う。他にもまだまだ改善する点はあると思うので、より良いプログラムが書けるようにこれからも勉強していきたいと思った。