

[Painel do utilizador](#) ▶ [As minhas unidades curriculares](#) ▶ [Programação](#) ▶ [Aulas práticas](#) ▶
[P01 08/03: syntax, primitive types, variables, expressions, input/output, choices, iteration](#) ▶

Início	quarta, 9 de março de 2022 às 08:59
Estado	Prova submetida
Data de submissão:	quarta, 9 de março de 2022 às 11:28
Tempo gasto	2 horas 29 minutos
Nota	100 do máximo 100

Pergunta 1

Correta Pontuou 20 de 20

Write a C++ program that "plays" a version of the known game *FizzBuzz* over a sequence of natural numbers up to an integer n provided by the user.

The program should build and finally print a string with each number in the sequence separated by a space. However:

- If the number is a multiple of 3, appends the word "Fizz" instead
- If the number is a multiple of 5, appends the word "Buzz" instead
- If the number is both a multiple of 3 and 5, nothing is done

Por exemplo:

Entrada	Resultado
7	1 2 Fizz 4 Buzz Fizz 7
11	1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11
16	1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11 Fizz 13 14 16
0	

Resposta: (regime de penalização: 0, 0, 0, 0, 10, 20, 30, ... %)

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     //cout << "Insira um número: ";
7     cin >> n;
8     string result;
9     for(int i = 1; i <= n; i++){
10         if(i%3 == 0 && i%5 == 0) continue;
11         if(i%3 == 0) {
12             result = result + "Fizz ";
13             continue;
14         }
15         if(i%5 == 0) {
16             result = result + "Buzz ";
17             continue;
18         }
19         result = result + to_string(i) + " ";
20     }
21     cout << result << endl;
22     return 0;

```

	Entrada	Esperado	Recebido	
✓	7	1 2 Fizz 4 Buzz Fizz 7	1 2 Fizz 4 Buzz Fizz 7	✓
✓	11	1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11	1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11	✓
✓	16	1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11 Fizz 13 14 16	1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11 Fizz 13 14 16	✓
✓	0			✓

Passou em todos os testes! ✓

Solução do autor da pergunta (Cpp):

```

1 #include <iostream>
2
3 using namespace std;
4
5 int main(void) {
6     int n;
7     // cout << "n? ";
8     cin >> n;
9     for (int i = 1; i <= n; i++) {
10         if (i % 3 == 0 && i % 5 == 0) continue;
11         if (i % 3 == 0) cout << "Fizz ";

```

```
12     else if (i % 5 == 0) cout << "Buzz ";
13     else cout << i << ' ';
14 }
15 cout << endl;
16 return 0;
17 }
18
19 // public tests (1 points each)
20 // 7 => "1 2 Fizz 4 Buzz Fizz 7"
21 // 11 => "1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11"
22 // 16 => "1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11 Fizz 13 14 16"
```

Correta

Nota desta submissão: 20/20

Pergunta 2

Correta Pontuou 20 de 20

Write a C++ program that, given an integer by user input, computes its reverse (the number with the digits by the reverse order) and prints it.

Por exemplo:

Entrada	Resultado
123	321
789	987
78989	98987
45654	45654

Resposta: (regime de penalização: 0, 0, 0, 0, 10, 20, 30, ... %)

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     string result;
8     while(n > 0){
9         int last_digit = n%10;
10        result = result + to_string(last_digit);
11        n = n/10;
12    }
13    cout << stoi(result) << endl;
14    return 0;
15 }
```

	Entrada	Esperado	Recebido	
✓	123	321	321	✓
✓	789	987	987	✓
✓	78989	98987	98987	✓
✓	45654	45654	45654	✓

Passou em todos os testes! ✓

Solução do autor da pergunta (Cpp):

```

1 #include <iostream>
2
3 using namespace std;
4
5 int main(void) {
6     int n, r = 0;
7     // cout << "n? ";
8     cin >> n;
9     while (n != 0) {
10        r = r * 10 + n % 10;
11        n = n / 10;
12    }
13    cout << r << endl;
14    return 0;
15 }
16
17 // public tests (1 points each)
18 // 123 => 321
```

```
19 // 789 => 987
20 // 78989 => 98987
21 // 45654 => 45654 // a palindrome!
22
```

Correta

Nota desta submissão: 20/20



Pergunta 3

Correta Pontuou 20 de 20

Write a C++ program that takes an integer n , provided by the user, and prints all prime numbers up to n .

A *prime number* (or a *prime*) is a natural number greater than 1 that can only be divided by 1 or itself [\[wikipedia\]](#).

Por exemplo:

Entrada	Resultado
20	2 3 5 7 11 13 17 19
19	2 3 5 7 11 13 17 19
2	2
1	

Resposta: (regime de penalização: 0, 0, 0, 0, 10, 20, 30, ... %)

```

1 #include <iostream>
2 using namespace std;
3
4 bool is_prime(int n){
5     if(n<2){
6         return false;
7     }
8     for(int i = 2; i < n; i++){
9         if(n%i == 0){
10            return false;
11        }
12    }
13    return true;
14 }
15
16 int main() {
17     int n;
18     cin >> n;
19     for(int i=2; i<=n; i++){
20         if(is_prime(i)){
21             cout << i << endl;
22         }
23     }
24 }
```

	Entrada	Esperado	Recebido	
✓	20	2 3 5 7 11 13 17 19	2 3 5 7 11 13 17 19	✓

	Entrada	Esperado	Recebido	
✓	19	2 3 5 7 11 13 17 19	2 3 5 7 11 13 17 19	✓
✓	2	2	2	✓
✓	1			✓

Passou em todos os testes! ✓

Solução do autor da pergunta (Cpp):

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main(void) {
6     int n;
7     // cout << "n? ";
8     cin >> n;
9     for (int i = 2; i <= n; i++) {
10         int j = 2;
11         while (j < i && i % j != 0) {
12             j++;
13         }
14         if (j == i) {
15             cout << j << endl;
16         }
17     }
18     return 0;
19 }
20
21 // public tests (1 points each)
22 // 20 => 2\n3\n5\n7\n11\n13\n17\n19
```

Correta

Nota desta submissão: 20/20

Pergunta 4

Correta Pontuou 20 de 20

In number theory, a [pernicious number](#) is a positive integer that has a prime number of **1s** in their binary representation. We are interested in numbers that are **both** prime and pernicious. For instance $7 = 111_2$ is both prime and pernicious, $6 = 110_2$ is pernicious but not prime, and $23 = 10111_2$ is prime but not pernicious.

Write a C++ program that prints the value of all numbers between **a** and **b** **that are both pernicious and prime**, where **a** and **b** are positive integers supplied as input.

Por exemplo:

Entrada	Resultado
10 50	11 13 17 19 31 37 41 47
11 47	11 13 17 19 31 37 41 47
1 11	3 5 7 11
23 83	31 37 41 47 59 61 67 73 79

Resposta: (regime de penalização: 0, 0, 0, 0, 10, 20, 30, ... %)

```

1 #include <iostream>
2 #include <algorithm>
3
4 using namespace std;
5
6 string toBinary(int n)
7 {
8     string r;
9     while(n!=0){
10         r = r + (n%2==0 ? "0":"1");
11         n/=2;
12     }
13     return r;
14 }
15
16 bool is_prime(int n){
17     if(n<2){
18         return false;
19     }
20     for(int i = 2; i < n; i++){
21         if(n%i == 0){
22             return false;

```


	Entrada	Esperado	Recebido	
✓	10 50	11 13 17 19 31 37 41 47	11 13 17 19 31 37 41 47	✓
✓	11 47	11 13 17 19 31 37 41 47	11 13 17 19 31 37 41 47	✓
✓	1 11	3 5 7 11	3 5 7 11	✓
✓	23 83	31 37 41 47 59 61 67 73 79	31 37 41 47 59 61 67 73 79	✓

Passou em todos os testes! ✓

Solução do autor da pergunta (Cpp):

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main(void) {
6      int a, b;
7      cin >> a >> b;
8      for (int i = a; i <= b; i++) {
9          int n = i, ones = 0;
10         while (n != 0) {
11             ones += n % 2;
12             n /= 2;
13         }
14         int j = 2;
15         while (j < ones && ones % j != 0){ j++; }
16         if (j == ones) {
17             // pernicious
18             while (j < i && i % j != 0){ j++; }
19             if (j == i) {
20                 // prime
21                 cout << i << endl;
22             }

```

Correta

Nota desta submissão: 20/20

Pergunta 5

Correta Pontuou 20 de 20

The value of Π can be approximated by the Gregory–Leibniz series:

$$4 \sum_{n=0}^k (-1)^n / (2n + 1)$$

Write a C++ program that given integers **k** and **d** prints the **k-th** approximation with **d** decimal places. You should make use of a value of type `double` for the approximation, and customise the print format using `std::fixed` and `std::setprecision` defined in header `<iomanip>`.

Por exemplo:

Entrada	Resultado
0 2	4.00
1 3	2.667
10 6	3.232316
100 7	3.1514934

Resposta: (regime de penalização: 0, 0, 0, 0, 10, 20, 30, ... %)

```

1 #include <iostream>
2 #include <math.h>
3 #include <iomanip>
4
5 using namespace std;
6
7 int main()
8 {
9     int k, d;
10    cin >> k >> d;
11
12    cout.precision(d);
13    cout << fixed;
14
15    double result;
16    for(int n=0; n<=k; n++){
17        result += pow(-1, n)/(2*n+1);
18    }
19    result *= 4;
20    cout << result << endl;
21    return 0;
22 }
```

	Entrada	Esperado	Recebido	
✓	0 2	4.00	4.00	✓
✓	1 3	2.667	2.667	✓
✓	10 6	3.232316	3.232316	✓
✓	100 7	3.1514934	3.1514934	✓

Passou em todos os testes! ✓

Solução do autor da pergunta (Cpp):

```

1 #include <iostream>
2 #include <iomanip>
3
4 int main() {
5     int k, d;
6     std::cin >> k >> d;
```

```
6  std::cin >> k >> u;
7  double approx = 0;
8  for (int i = 0; i <= k; i++) {
9      double term = 1.0 / (2 * i + 1);
10     approx += i % 2 == 0 ? term : -term;
11 }
12 approx *= 4;
13 std::cout << std::fixed << std::setprecision(d) << approx << std::endl;
14 return 0;
15 }
16
17 // public tests (1 points each)
18 // 0 2 => 4.00
19 // 1 3 => 2.667
20 // 10 6 => 3.232316
21 // 100 7 => 3.1514934
22
```

Correta

Nota desta submissão: 20/20

[◀ Development environment - setup instructions \(GitHub\)](#)

Ir para...

T01 08/03 ▶