Universidade Federal de Ouro Preto – UFOP
Instituto de Ciências Exatas e Biológicas – ICEB
Departamento de Computação – DECOM
Disciplina: BCC201 – Introdução a Programação
Professores: Túlio A. M. Toffolo, Puca Huachi V. Penna e Alan R. R. de Freitas

Aula Prática P-06

* Todos os exercícios que envolvem programas devem ser resolvidos através de programas em C/C++.
* A entrega será feita até às 23h55 do dia da aula prática no Moodle, sem zipar (entregue apenas o código fonte)
* Inclua seu número de matrícula, nome e turma em um comentário no início de cada arquivo com código fonte.
* Você só pode utilizar conhecimento prévios à aula para resolver o exercício. Caso use uma matéria que ainda não foi dada sua nota será penalizada.
* Códigos que não compilam serão zerados.

Questão 01

O valor aproximado do número π pode ser calculado por meio da seguinte forma: $\pi = \sqrt[3]{S \times 32}$, onde S é dada pela série:

$$S = 1 - \frac{1}{3^3} + \frac{1}{5^3} - \frac{1}{7^3} + \frac{1}{9^3} - \cdots$$

Codifique um programa para ler o número termos (quanto maior, melhor a precisão), calcular e imprimar o valor de π.

O cálculo deve ser feito por uma função que recebe o número de termos como parâmetro e retorna o valor de π.

Questão 02

Quando se está trabalhando em um sistema corporativo, é comum a necessidade de validar CPF. Para o CPF ser válido não basta apenas atender à máscara "###.###.### – ##" (onde o caractere '#' representa um número). Existe uma regra matemática que também deve ser verificada para um CPF ser considerado válido.

Faça um programa que leia um CPF (**somente números**) e verifique se ele é válido. O cálculo deve ser feito em uma função, que recebe o CPF e retorna 1, se for válido e 0, caso contrário. O programa e a função deverão usar o tipo `long` para armazenar o CPF.

É obrigatório o uso de comandos de repetição na separação dos algarismos e no cálculo da verificação.

O cálculo para validar um CPF é especificado pelo Ministério da Fazenda. Vamos entender como funciona:

O CPF é formado por 11 dígitos numéricos que seguem a máscara "###.###.### − ##", a sua verificação acontece utilizando os 9 primeiros dígitos e, com um cálculo simples, verificando se o resultado corresponde aos dois últimos dígitos (depois do símbolo '−').

Note que, para efetuar cálculos com estes dígitos, o programa deve se utilizar de operações matemáticas para identificar estes algarismos dentro do número completo do tipo `long`.

Vamos usar como exemplo, um CPF fictício `52998224725`.

Validação do primeiro dígito

1. Inicialmente, multiplicam-se os 9 primeiros dígitos pela sequência decrescente de números de 10 à 2 e somam-se os resultados. Exemplo:

   $5 \times 10 + 2 \times 9 + 9 \times 8 + 9 \times 7 + 8 \times 6 + 2 \times 5 + 2 \times 4 + 4 \times 3 + 7 \times 2 = 295$

2. No próximo passo da verificação basta multiplicar esse resultado por 10 e pegar o resto da divisão inteira por 11: $295 \times 10/11 = 268$ e o RESTO é '2'

3. Se o resto for igual ao primeiro dígito verificador (primeiro dígito depois do '−'), a primeira parte da validação está correta. Isso significa que o nosso CPF exemplo passou na validação do primeiro dígito.

Observação Importante: Se o resto da divisão for igual a 10, nós o consideramos como 0.

Validação do segundo dígito

1. Na validação do segundo dígito, vamos considerar os 9 primeiros dígitos, mais o primeiro dígito verificador, e vamos multiplicar esses 10 números pela sequência decrescente de 11 a 2. Vejamos: $5 \times 11 + 2 \times 10 + 9 \times 9 + 9 \times 8 + 8 \times 7 + 2 \times 6 + 2 \times 5 + 4 \times 4 + 7 \times 3 + 2 \times 2 = 347$

2. Seguindo o mesmo processo da primeira verificação, multiplicamos por 10 e pegamos o resto da divisão 11: $347 \times 10/11 = 315$ e o RESTO é '5'.

3. Finalmente, comparamos o resto (5), com o segundo dígito verificador (5). Com essa verificação, constatamos que o CPF nº `529.982.247-25` é válido.

Questão 03

Desde a aula de ontem, *Bart Simpson* continua tentando aprender a jogar xadrez. Ele aprendeu como uma Torre se move, mas tem dificuldade em saber para qual direção ele pode mover um **Bispo**. Sabendo que um tabuleiro de xadrez é composto por 8 linhas e 8 colunas, e que o **Bispo** se move nas diagonais:

- Escreva um programa que solicite ao *Bart* o número da linha e da coluna que indicam a posição do **Bispo**. O programa deve imprimir quais são os possíveis movimentos.
- Utilize `"-"` para indicar uma casa para a qual o **Bispo** não pode ser movido e `"x"` para indicar uma casa para a qual ele pode ser movido. Para indicar a posição do bispo use `"o"` .

Exemplo de execução (valores digitados pelo usuário destacados em azul):

```
Movimentos de um Bispo no xadrez!
Digite a linha  em que o Bispo se encontra: 6
Digite a coluna em que o Bispo se encontra: 3

Movimentos possíveis:

      1  2  3  4  5  6  7  8
    ------------------------
 1 |  -  -  -  -  -  -  -  x
 2 |  -  -  -  -  -  -  x  -
 3 |  -  -  -  -  -  x  -  -
 4 |  x  -  -  -  x  -  -  -
 5 |  -  x  -  x  -  -  -  -
 6 |  -  -  o  -  -  -  -  -
 7 |  -  x  -  x  -  -  -  -
 8 |  x  -  -  -  x  -  -  -
```