

Universidade Federal do Piauí
Campus Senador Helvídio Nunes de Barros - Picos
Disciplina: Algoritmos e Programação II
Curso: Bacharelado em Sistemas de Informação
Professor: Glauber Dias Gonçalves

Lista de Atividades 03: Funções e Modularização

Todas as questões abaixo devem ser implementadas na linguagem C

1. Escreva uma função que recebe um valor inteiro e retorna o número com seus dígitos invertidos. Por exemplo, dado o número 7631, a função deve retornar 1367. Escreva um programa que recebe como entrada um número inteiro de 1000 a 9999 e utilize a função.

2. Escreva uma função *distância* que calcule a distância entre dois pontos. A função recebe como parâmetros os valores $x1$ e $y1$, representando ponto1, e os valores $x2$ e $y2$ representando ponto2. Todos os parâmetros de entrada e o retorno da função devem ser do tipo *float*. Escreva um programa que recebe os quatro números da entrada padrão representando respectivamente $x1$, $y1$, $x2$ e $y2$ e utiliza a função *distância* para calcular a distância entre os pontos. Funções da biblioteca *math.h*, por exemplo *sqrt*, podem ser utilizadas na função.

3. Escreva uma função que simule o lançamento de uma moeda (use a função *rand* da biblioteca *stdlib.h*). A função deve retornar 0 para coroa e 1 para cara. Escreva um programa para lançar a moeda 100 vezes. Para cada lançamento da moeda, o programa deve imprimir "cara" ou "coroa" e, ao final dos lançamentos, imprimir o número de vezes que o lado "cara" apareceu.

Nota: Se o programa simular realisticamente o lançamento da moeda, o número de vezes deve ser aproximadamente metade dos lançamentos, ou seja, aproximadamente 50.

4. Um dos jogos de azar mais populares é um jogo de dados conhecido como "craps", que é jogado em cassinos em todo o mundo. As regras do jogo são as seguintes:

Um jogador rola dois dados. Cada dado tem seis faces. Estas faces contêm 1, 2, 3, 4, 5 e 6 pontos. Assim que os dados descansam, a soma dos pontos nas duas faces no topo dos dados é calculado. Se a soma for 7 ou 11 no primeiro lançamento, o jogador ganha. Se a soma for 2, 3 ou 12 no primeiro lançamento (chamado "craps"), o jogador perde (ou seja, a "casa" ganha). Se a soma é 4, 5, 6, 8, 9 ou 10 no primeiro lançamento, então essa soma se torna "Ponto" do jogador. Para ganhar, você deve continuar rolando os dados até que você repita novamente seu ponto. O jogador perde rolando um 7 antes de fazer o ponto.

Escreva um programa modularizado que simula o jogo de "craps".

Veja exemplos da tela de saída da simulação:

Soma dos dados $5 + 6 = 11$

O jogador ganhou :)

Soma dos dados $4 + 1 = 5$

O ponto é 5

Soma dos dados $6 + 2 = 8$

Soma dos dados $2 + 1 = 3$

Soma dos dados $3 + 2 = 5$
O jogador ganhou :)

Soma dos dados $1 + 1 = 2$
O jogador perdeu :(

Soma dos dados $6 + 4 = 10$
O ponto é 10
Soma dos dados $3 + 4 = 7$
O jogador perdeu :(

5. Escreva um programa modularizado que reproduz o jogo de “adivinha o número” da seguinte maneira: seu programa escolhe o número a ser adivinhado selecionando um número inteiro aleatoriamente no intervalo de 1 a 1000. O programa deve funcionar da seguinte forma:

Eu tenho um número entre 1 e 1000.

Você pode adivinhar meu número?

Digite seu primeiro palpite: << o jogador faz uma adivinhação e o programa responde com >>

1. Excelente! Você adivinhou o número!
2. Muito baixo. Tente novamente.
3. Muito alto. Tente novamente.

Se o palpite do jogador estiver incorreto, o programa deve repetir até que o jogador finalmente obtenha o número certo. Seu programa deve continuar informando “Muito alto” ou “Muito baixo” para ajudar o jogador ir na resposta correta. Ao encontrar a resposta o programa deve imprimir o número de tentativas do jogador. [Nota: A técnica de busca empregada nesse problema é chamada de pesquisa binária.]

6. Escreva um programa modularizado que use geração de números aleatórios para criar frases. O programa deve usar quatro matrizes de caracteres: artigo, substantivo, verbo e preposição. O programa deve criar uma frase selecionando uma palavra aleatoriamente de cada matriz na seguinte ordem: artigo, substantivo, verbo, preposição, artigo e substantivo. Enfim, o programa deverá concatenar às palavras em um vetor que seja suficientemente grande para conter a frase inteira. As palavras devem ser separadas por espaços. A frase final deve ser impressa na tela iniciando com uma letra maiúscula e terminando com um ponto. O programa deve gerar 20 frases aleatórias. As matrizes devem ser preenchidas como se segue:

- A matriz artigo deve conter "o", "um", "algum", "todo" e "qualquer";
- A matriz substantivo deve conter "menino", "homem", "cachorro", "carro", "gato";
- A matriz verbo deve conter "passou", "pulou", "correu", "saltos", "andou";
- A matriz preposição deve conter "sobre", "sob", "ante", "ate", "com".