

<b>CURSO:</b> Segurança da Informação		<b>DISCIPLINA:</b> Programação I		<b>Turma</b> SEG NA1	
<b>DOCENTE:</b> Alan Henrique Pardo de Carvalho					
<b>AVALIAÇÃO OFICIAL:</b> <input type="checkbox"/> N1 <input checked="" type="checkbox"/> N2 <input type="checkbox"/> N3				<b>DATA:</b> 08/06/2023	
<b>NOMES DOS ALUNOS:</b>		Anselmo Antunes Ribeiro			
<p><b>Condições:</b> Consulta aos materiais é permitida. Essa prova pode ser feita individualmente ou em duplas. Se for feita em duplas, apenas um integrante da dupla precisa entregar a prova no Teams, mas os NOMES COMPLETOS dos dois alunos da dupla precisarão estar no campo acima. A compreensão das questões faz parte da prova. Nas questões onde seja necessário marcar uma alternativa, basta clicar na <i>checkbox</i> correspondente para ativá-la ou desativá-la. Salve esse arquivo preenchido em formato PDF e entregue o PDF na tarefa do Teams relativa à prova. Você pode usar qualquer ambiente para escrever os programas, mas tenha em mente que eles poderão ser testados e validados no Dev C++.</p> <p><b>Horário máximo para entrega:</b> até as 23h59 do dia 11/06/2023. O Teams fechará automaticamente a tarefa e não haverá como entregar depois.</p>					

1. Marque a caixa correspondente a “SIM” ou “NÃO” (só pode ser uma das alternativas) para cada afirmação a seguir. Cada afirmação com a marcação correta vale 0,1 ponto.

AFIRMAÇÃO	SIM	NÃO
Caso você tenha de verificar se o valor da variável <b>a</b> está entre 10 e 20, a instrução <b>switch</b> é a mais apropriada.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
A instrução <b>for</b> é mais comumente utilizada quando não se sabe a quantidade de repetições em um loop.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Um array é uma forma simples de armazenar diversos valores de tipos diferentes sob o mesmo nome, ao invés de utilizar diversas variáveis com nomes diferentes.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Se você criar o array <b>numero[5]</b> , para se referir ao quarto elemento do array deverá usar <b>numero[3]</b> .	<input checked="" type="checkbox"/>	<input type="checkbox"/>
A instrução <b>do...while</b> é mais comumente utilizada quando não sabemos se a sequência de instruções do loop deverá ser executada pelo menos uma vez.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Dependendo do que aconteça antes que a instrução <b>while</b> seja executada, é possível que a sequência de instruções do loop não seja executada nem uma vez sequer.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Uma variável do tipo <b>char</b> pode ter armazenada uma palavra como “computador”.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Uma função que não tem valor de retorno deve ser declarada como sendo do tipo <b>void</b> .	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Não é possível criar na função <b>main()</b> a variável <b>int valor</b> e dentro da função <b>calcula()</b> a variável <b>float valor</b> pois elas, além de terem o mesmo nome, são de tipos diferentes.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Para uma função retornar dois valores, deve-se usar <b>return valor1, valor2</b> .	<input type="checkbox"/>	<input checked="" type="checkbox"/>

2. Por que no seguinte programa a biblioteca **ctype.h** é necessária? (1,0 ponto)

```
#include <stdio.h>
#include <locale.h>
#include <conio.h>
#include <ctype.h>

int main() {
    setlocale(LC_ALL, "");
    char mais;
    printf("Deseja continuar? (S/N) ");
    mais = toupper(getche());
    if (mais == 'S') {
        printf("\nVamos continuar.\n");
    }
    else {
        printf("\nAcabou.\n");
    }
    return 0;
}
```

O uso da biblioteca 'ctype.h' é necessário para possibilitar o uso da função 'toupper', essa função permite transformar em maiúsculas letras minúsculas e quando o usuário digitar algo diferente de uma letra, o que foi digitado se mantém inalterado. Nesse programa essa função é importante para que não aconteça um resultado indesejado na compilação do programa, caso o usuário digite 's' em vez de 'S'.

3. Uma escola concede um desconto na mensalidade aos seus alunos de acordo com o tipo de matrícula que cada um fez. Crie um programa que peça os dados necessários e exiba o valor do desconto a ser dado na mensalidade, bem como o valor da mensalidade com desconto. A tabela a seguir mostra os percentuais de desconto de acordo com o tipo de matrícula feita. Nesse programa deve ser utilizada a instrução **switch**. (2,0 pontos)

TIPO DE MATRÍCULA	DESCONTO SOBRE A MENSALIDADE
1	3%
2	4%
3	5%
Outros	0

```
#include <stdio.h>
#include <locale.h>

int main(){
    setlocale(LC_ALL, "");

    int tipo_matricula;
    float desconto, calculo_desconto, mensalidade, valor_total;

    printf("Digite o valor da mensalidade:");
    scanf("%f", &mensalidade);
    printf("Digite o tipo de Matrícula\n1 - Matrícula1\n2 - Matrícula2\n3 - Matrícula\n4 - Outro\n==>");
    scanf("%d", &tipo_matricula);

    switch (tipo_matricula){
        case 1: desconto = 0.03; break;
        case 2: desconto = 0.04; break;
        case 3: desconto = 0.05; break;
        default: desconto = 0;
    }

    calculo_desconto = mensalidade * desconto;

    valor_total = mensalidade - calculo_desconto;

    printf("Desconto obtido: R$ %.2f", calculo_desconto);
    printf("\nValor com desconto a pagar: R$ %.2f", valor_total);

    return 0;
}
```

4. Complete o código fonte a seguir para que o programa faça a validação na entrada de dados, sendo que não pode ser aceito um produto com preço menor que R\$ 0,01 e deve aceitar somente a entrada

dos tipos de produtos informados na mensagem. Tanto para o preço como para o tipo do produto, caso valores inválidos sejam informados, a respectiva mensagem de erro deverá ser exibida e o dado deverá ser solicitado novamente, quantas vezes forem necessárias de forma que a mensagem “Vamos continuar.” somente será exibida depois que um produto com preço e tipo válidos for informado. (2,0 pontos)

```
#include <stdio.h>
#include <locale.h>

int main() {
    setlocale(LC_ALL, "");

    float preco_unitario;
    char tipo;

    printf("Digite o preço unitário do produto: ");
    scanf("%f", &preco_unitario);
    printf("Digite o tipo do produto:\na - essencial\n");
    printf("b - importante\nc - supérfluo ==> ");
    tipo = getche();
    printf("\nVamos continuar.");

    /* daqui para baixo haveria mais código, dependendo do que
       o programa faça, mas isso não é relevante para essa
       questão da prova
    */

    return 0;
}
```

```
#include <stdio.h>
#include <locale.h>
#include <conio.h>

int main() {
    setlocale(LC_ALL, "");

    float preco_unitario;
    char tipo, a, b, c;

    printf("Digite o preço unitário do produto: ");
    scanf("%lf", &preco_unitario);

    while("%f", preco_unitario < 0.01){
        printf("O valor deve ser maior que R$0.01, insira novamente:");
        scanf("%lf", &preco_unitario);
    }

    printf("Digite o tipo do produto:\na - essencial\n");
    printf("b - importante\nc - supérfluo ==> ");

    while(1){
        tipo = getche();
        if(tipo == 'a' || tipo == 'b' || tipo == 'c'){
            break;
        }
        printf("\nTipo precisa ser a, b ou c, digite novamente:");
    }
    printf("\nVamos continuar.");

    return 0;
}
```

5. Crie um programa que peça ao usuário dez números inteiros e os armazene em um array. Após pedir os dez números, o programa deverá exibir:

- A lista de números digitados, um ao lado do outro e separados por um espaço, do primeiro ao último número.
- Abaixo da lista de números digitados, a soma desses números.
- Abaixo da soma dos números, a média desses números com duas decimais.
- Abaixo da soma dos números, a quantidade de números pares digitados.
- Abaixo da quantidade de números pares digitados, a lista de números digitados, um ao lado do outro e separados por um espaço, do último ao primeiro número. (2,0 pontos)

A seguir, temos uma tela de exemplo com a forma como os valores devem ser inseridos e exibidos. Obviamente, o programa precisará exibir os resultados corretos para quaisquer números inteiros inseridos.

```
Questão 5 da prova

Digite um número: 12
Digite um número: 48
Digite um número: 37
Digite um número: -8
Digite um número: 5
Digite um número: 25
Digite um número: 33
Digite um número: 44
Digite um número: 88
Digite um número: 100

Lista de números digitados: 12 48 37 -8 5 25 33 44 88 100
Soma dos números digitados: 384
Média dos números digitados: 38,40
Quantidade de números pares digitados: 6
Lista de números digitados ao contrário: 100 88 44 33 25 5 -8 37 48 12
```

```
#include <stdio.h>
#include <locale.h>

int main(){
    setlocale(LC_ALL, "");

    int num[10], a, soma = 0, pares = 0;
    float media;

    for(a = 0; a < 10; a++){
        printf("Digite os números\n");
        scanf("%d", &num[a]);
    }

    printf("Números digitados:");

    for(a = 0; a < 10; a++){
        printf("%d ", num[a]);
        soma += num[a];
        if (num[a] % 2 == 0) {
            pares++;
        }
    }

    printf("\nSoma dos números: %d", soma);
    media = (float)soma / 10;
    printf("\nMédia dos números: %.2f", media);
    printf("\nQuantidade de números pares: %d", pares);
    printf("\nNúmeros digitados (do último ao primeiro): ");
```

```

    for (a = 9; a >= 0; a--) {
        printf("%d ", num[a]);
    }

    return 0;
}

```

6. Crie um programa que peça ao usuário dez números inteiros e os armazene em um array. Após pedir os dez números, o programa deverá exibir os números digitados, um abaixo do outro, mas apenas enquanto não for encontrado o número 0, devendo parar e exibir a mensagem “Fim da lista”. A tela a seguir mostra um exemplo com uma sequência de números informada e como devem ser exibidos os números e a mensagem final. Obviamente, o programa precisará funcionar corretamente independentemente da sequência informada. (2,0 pontos)

```

Questão 6 da prova

Digite um número: 10
Digite um número: 20
Digite um número: 30
Digite um número: 40
Digite um número: 50
Digite um número: 0
Digite um número: 70
Digite um número: 80
Digite um número: 90
Digite um número: 100

10
20
30
40
50

Fim da lista.

```

```

#include <stdio.h>
#include <locale.h>

int main(){
    setlocale(LC_ALL, "");

    int num[10], a, soma = 0;
    float media;

    for(a = 0; a < 10; a++){
        printf("Digite os números\n");
        scanf("%d", &num[a]);
    }

    printf("Números digitados:\n");

    for(a = 0; a < 10; a++){
        if(num[a] == 0){
            printf("Fim da lista");
            break;
        }
        printf("%d\n", num[a]);
    }

    return 0;
}

```

}