

LISTA I (Revisão)

1. Qual o valor de x, y e p no final da execução desse trecho de código?

```
int x, y, *p;
y = 0;
p = &y;
x = *p;
x = 4;
(*p)++;
x--;
(*p) += x;
```

2. Qual é o resultado da execução desse programa?

```
void imprime_primeiro(int *vet)
{
    printf("Valor: %d\n", vet[0]);
}

int main(void)
{
    int vet[5] = {1, 2, 3, 4, 5};
    imprime_primeiro(vet);
    return 0;
}
```

3. Qual é o resultado da execução desse programa?

```
void imprime_primeiro(int *vet)
{
    printf("Valor: %d\n", vet[0]);
}

int main(void)
{
    int vet[5] = {1, 2, 3, 4, 5};
    imprime_primeiro(&vet[2]);
    return 0;
}
```

4. Qual é o resultado da execução desse programa?

```
int* metade_final(int *vet, int n)
{
    return &vet[(int)(n/2)];
}

int main(void)
{
    int vet[6] = {1, 2, 3, 4, 5, 6};
    int *v = metade_final(vet, 6);
    printf("Valor: %d\n", v[0]);
    return 0;
}
```

5. Qual é o resultado da execução desse programa?

```
int main(void)
{
    int vet[6] = {1, 2, 3, 4, 5, 6};
    printf("Valor1: %d\n", vet);
    printf("Valor2: %d\n", *vet);
    printf("Valor3: %d\n", *(vet + 2));
    return 0;
}
```

6. Implemente uma função que busca por um inteiro em um vetor de inteiros.
7. Implemente uma função que busca por um inteiro em um vetor de inteiros. Retornando a última ocorrência.
8. Criar uma função para encontrar o valor máximo em um vetor.
9. Criar uma função para encontrar o valor mínimo em um vetor.
10. Faça um programa para ler e armazenar um conjunto de números em um vetor (máximo 50 números e a leitura de um número igual a zero indica fim da leitura dos dados). A seguir peça para o usuário digitar um valor inteiro e informe se o mesmo pertence ou não ao conjunto de números armazenados. O programa deve implementar e usar a função busca, que recebe como parâmetro um vetor de números inteiros (vet) de tamanho n e um valor x. A função deve retornar 1 se x pertence a esse vetor e 0 caso contrário. A função deve obedecer ao seguinte protótipo:

```
int busca(int vet[], int n, int x);
```

11. Faça um programa para ler uma matriz digitada pelo usuário (tamanho 3x3) e exibir uma mensagem dizendo se a mesma é uma matriz identidade ou não. O seu programa deve implementar e utilizar a função matriz_identidade, que recebe como parâmetro uma matriz quadrada de inteiros de dimensão n, e retorna 1 se a matriz for uma matriz identidade, e 0 caso contrario. A função deve obedecer ao seguinte protótipo:

```
int matriz_identidade(int mat[][N], int n);
```

Lembre-se que uma matriz é considerada identidade quando os elementos da diagonal principal são todos iguais a 1 e os elementos restantes são iguais a zero.