

Par x Ímpar

Assim como o conjunto dos números naturais é subconjunto dos números inteiros, o conjunto dos números pares também é. Em um primeiro momento, aprendemos a reconhecer os elementos do conjunto dos números pares por meio de brincadeiras. A regra usada é: todo número par termina com 0, 2, 4, 6 ou 8. Dessa maneira, 224, por exemplo, é um número par porque termina com o algarismo 4.

Entretanto, essa é uma consequência da definição formal de número par, que pode ser compreendida como: todo número par é múltiplo de 2.

Existem outras definições para os elementos desse subconjunto dos números inteiros, por exemplo: todo número par é divisível por 2.

A “definição algébrica” usada para reconhecer os elementos desse conjunto é: dado um número p , pertencente ao conjunto dos números inteiros, p será par se: $p = 2n$. Nesse caso, n é um elemento do conjunto dos números inteiros. Note que essa é a “tradução” da primeira definição em termos algébricos.

Já números ímpares são os elementos do conjunto dos números inteiros que não são pares, ou seja, são os números que terminam com algum dos algarismos 1, 3, 5, 7 ou 9. Formalmente, o conjunto dos números ímpares é um subconjunto dos números inteiros, e a definição de seus elementos é: todo número ímpar não é múltiplo de 2.

Atividade

Criar um programa na linguagem C para separar uma sequência de números inteiros fornecidos pelo usuário em pares e ímpares.

Funcionamento

Primeiramente, o usuário deverá fornecer a quantidade de números da sequência e, posteriormente, os números, propriamente, dito.

Restrições:

- O tamanho máximo da sequência fornecida pelo usuário é 100.
- Somente números inteiros e positivos deverão ser aceitos.

Dicas:

- O operador % retorna o resto de uma divisão. Se o resto de uma divisão for 0, o número é par, caso contrário, ímpar;

- Crie três vetores: 1) entrada de dados, 2) para os números pares e 3) para os ímpares.

Exemplo 1

Entrada:

```
4
1
4
89
56
```

Saída:

```
Pares .: 4 56
Impares: 1 89
```

Exemplo 2

Entrada:

```
8
45
87
9
63
58
32
54
89
```

Saída:

```
Pares .: 58 32 54
Impares: 45 87 9 63 89
```

Exemplo 3

Entrada:

```
20
8
9
4
5
56
8
9
99
65
```

```
65
98
658
45
5
6
3
21
20
58
3
```

Saída:

```
Pares .: 8 4 56 8 98 658 6 20 58
Impares: 9 5 9 99 65 65 45 5 3 21 3
```

Exemplo 4

Entrada:

```
5
9
5
9
99
65
```

Saída:

```
Pares .:
Impares: 9 5 9 99 65
```

Exemplo 5

Entrada:

```
6
4
2
90
88
6
42
```

Saída:

```
Pares .: 4 2 90 88 6 42
Impares:
```

Exemplo 6

Entrada:

1
2

Saída:

Pares .: 2
Impares:

Exemplo 7

Entrada:

3
0
1
2

Saída:

Pares .: 0 2
Impares: 1

Autotest

1. É importante que seu programa siga exatamente as instruções acima e passe em todos os testes realizado pelo **autotest**, caso contrário, mesmo estando correto, você perderá pontos.
2. Passar no **autotest** não é garantia de receber todos os pontos. A avaliação final é do professor, que irá avaliar a lógica, a estrutura e a estética do programa. Também irá verificar se houve ou não plágio.
3. Comando para o **autotest**:

```
autotest.exe ..\bin\Debug\par_impar.exe
```

Há um espaço entre o **exe** e os dois pontos ..