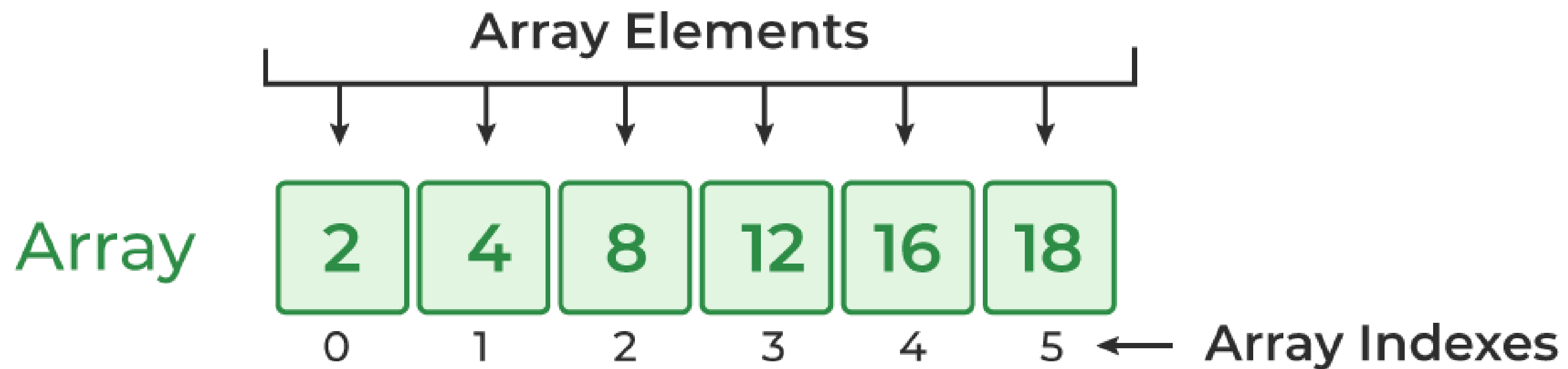


Fundamentos de Algoritmos

Introdução a vetores

DESVENDANDO OS VETORES

O QUE SÃO VETORES ?



VANTAGENS DOS VETORES

Eficiência – Economize tempo e espaço na memória ao armazenar vários dados em um único lugar.

Organização: Mantenha seus dados organizados e de fácil acesso.

Agilidade: Acesse qualquer dado do vetor com rapidez e precisão.

Declarando Vetores




```
1  tipo_de_dado nome_do_vetor[tamanho];
```

INICIALIZANDO VETORES

Após criar o baú, você precisa preenchê-lo com seus itens mágicos.
Existem três formas de fazer isso

INDIVIDUALMENTE



```
1 moedas_ouro[0] = 1;  
2 moedas_ouro[1] = 2;  
3 ...  
4 moedas_ouro[9] = 10;
```

INICIALIZANDO VETORES

Após criar o baú, você precisa preenchê-lo com seus itens mágicos.
Existem três formas de fazer isso

EM BLOCO



```
1 int moedas_ouro[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

INICIALIZANDO VETORES

Após criar o baú, você precisa preenchê-lo com seus itens mágicos.
Existem três formas de fazer isso

COM UM VALOR ÚNICO



```
1 char feiticos[50] = {'F'}; // Todos os feitiços são "Fogo" no início.
```

INICIALIZANDO VETORES

Após criar o baú, você precisa preenchê-lo com seus itens mágicos.
Existem três formas de fazer isso

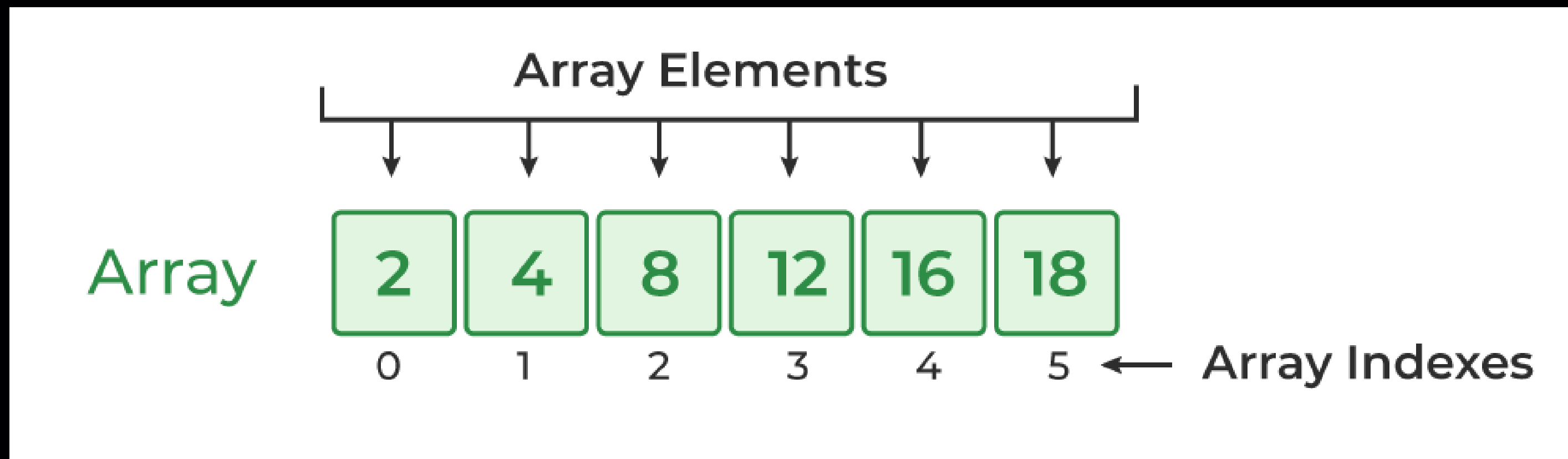
COM UM VALOR ÚNICO



```
1 char feiticos[50] = {'F'}; // Todos os feitiços são "Fogo" no início.
```

ACESSANDO OS ELEMENTOS

Cada item do vetor possui um número único, chamado de índice. Para acessar um item específico, basta usar o índice como chave para abrir o baú:



ACESSANDO OS ELEMENTOS

Cada item do vetor possui um número único, chamado de índice. Para acessar um item específico, basta usar o índice como chave para abrir o baú:



```
1 int moeda_escolhida = moedas_ouro[3]; // Pega a 4ª moeda (índice 3).  
2 char primeiro_feitico = feiticos[0]; // Descobre o primeiro feitiço.
```

EXEMPLO PRÁTICO:

```
1  #include <stdio.h>
2
3  int main() {
4      // Declara um array de 5 inteiros
5      int numeros[5] = {10, 20, 30, 40, 50};
6
7      // Acessa e imprime o primeiro elemento do array
8      printf("Primeiro elemento: %d\n", numeros[0]);
9
10     // Acessa e imprime o terceiro elemento do array
11     printf("Terceiro elemento: %d\n", numeros[2]);
12
13     // Altera o valor do quarto elemento do array
14     numeros[3] = 60;
15
16     // Imprime o valor atualizado do quarto elemento
17     printf("Quarto elemento atualizado: %d\n", numeros[3]);
18
19     // Percorre o array e imprime cada elemento
20     for (int i = 0; i < 5; i++) {
21         printf("Elemento %d: %d\n", i, numeros[i]);
22     }
23
24     return 0;
25 }
26
```

DESAFIO

Cria um baú mágico para armazenar 5 poções mágicas (strings).

Preencha o baú com 5 poções diferentes ("Cura", "Força", "Invisibilidade", "Teletransporte" e "Visão Noturna").

Mostre o nome de cada poção no baú.

Pergunte ao usuário qual poção ele deseja usar e mostre seus poderes.

