Exercícios

#include <iostream>

int main(int argc, char\*\* argv) {

char nome[40], tipo[40];

int vida, ataque, defesa;

fflush(stdin);

printf("\nDigite o nome do seu pokemon: ");

gets(nome);

printf("\nDigite o tipo do seu pokemon: ");

gets(tipo);

printf("\nDigite a vida do pokemon: ");

scanf("%d", &vida);

printf("\nDigite o ataque do pokemon: ");

scanf("%d", &ataque);

printf("\nDigite a defesa do pokemon: ");

scanf("%d", &defesa);

printf("\nNome: %s", nome);

printf("\nTipo: %s", tipo);

printf("\nVida: %d", vida);

printf("\nAtaque: %d", ataque);

printf("\nDefesa: %d", defesa);

return 0;

}

1. Observe o código acima atentamente. Agora, altere o código para que todas as variáveis sejam agrupadas em uma struct e adapte o código para que ele possa ser executado. Quando funcionar, gaste um tempo para perceber como a função main ficou mais organizada.
2. Ainda com o código do início do exercício, altere-o para que possua dois pokémons (não use structs!!!). Então, deve pedir os valores para os dois pokémons e, por fim, mostrar os valores dos 2 pokemons.
3. Agora, pegue o código do exercício 2 e altere-o, usando structs. Depois que conseguir executar, gaste um tempo observando como este código ficou mais organizado que o do exercício 2. Ficou ainda mais evidente a organização das variáveis, certo?
4. Seguindo a evolução do código, agora crie duas funções, uma para cada tarefa do código (uma para pedir os dados de um pokémon e uma para apresentar). Crie as funções como sempre fez.
5. Agora, altere o código para que as funções pertencentes ao pokémon sejam criadas de dentro das structs. Consegue perceber uma melhoria na organização? É sutil, ainda.
6. Por fim, pegue o código do exercício 4. Vamos melhorar essa percepção sobre o que observamos no exercício anterior a seguinte forma: crie, também, no código do exercício 4, crie uma struct para representar treinador. Um treinador deve ter nome, cidade de origem e idade. Além das funções existentes para os pokémons, Crie uma função para ler os valores de um treinador e uma função para apresentar os valores digitados.
7. Agora, assim como no exercício 5, adapte o código para que as funções relacionadas ao pokémon sejam definidas dentro da struct pokémon e as funções relacionadas ao treinador sejam definidos dentro da struct do treinador. O código ficou mais organizado!!! Observe isso!!!