



UNIVERSIDADE FEDERAL DE MINAS GERAIS

Departamento de Ciência da Computação

TRABALHO PRÁTICO SIMULADOR DE BATALHA POKÉMON

LUIS HENRIQUE EMEDIATO DE REZENDE LARA

Junho de 2024

UNIVERSIDADE FEDERAL DE MINAS GERAIS.....	1
TRABALHO PRÁTICO.....	1
1. Introdução.....	3
2. Descrição do Algoritmo e Procedimentos utilizados.....	4
2.1. Estrutura geral.....	4
2.2. Exemplo de Estilo texto esperado.....	6
2.3. Exemplo de execução.....	7
3. Testes e Erros.....	8
3.1. Processo de Criação do Algoritmo.....	8
3.2. Exemplos de Erros Comuns e Correções.....	8
3.3. Testes Realizados.....	8
4. Conclusão.....	15

1. Introdução

A atividade desenvolvida no trabalho prático consiste em implementar um programa na linguagem C para simular uma batalha Pokémon entre dois treinadores. A batalha é composta por uma série de lutas entre os Pokémon de dois treinadores, que continua até que um dos treinadores não tenha mais Pokémon com vida, ocasionando assim, o encerramento da batalha.

O programa deve ler um arquivo .txt contendo informações sobre a quantidade de Pokémons de cada treinador e suas características (nome, ataque, defesa, vida e tipo do Pokémon). Em seguida, deve simular a batalha entre os treinadores e, ao final, exibir o resultado para o usuário. Isso inclui imprimir os dados do arquivo .txt, os resultados de cada luta, os Pokémons sobreviventes e derrotados, além do treinador vencedor.

É importante notar que, neste contexto, "batalha" refere-se ao confronto geral entre os treinadores, enquanto "luta" refere-se ao confronto individual entre dois Pokémons e "treinador" refere-se a cada um dos dois jogadores da simulação. Portanto, o objetivo final do programa é declarar o jogador vencedor com base na simulação das batalhas entre seus Pokémons.

2. Descrição do Algoritmo e Procedimentos utilizados

2.1. Estrutura geral

Após incluir as bibliotecas necessárias (stdio.h, stdlib.h, string.h) no programa, foi criada uma estrutura intitulada "POKEMON". Esta estrutura contém cinco variáveis que representam os seguintes atributos de um Pokémon:

1. Nome: um vetor de caracteres para o nome (nome[20]), foi adicionado um espaço extra para armazenar os nomes dos Pokémon que, em geral, podem ter até 15 caracteres, totalizando 20 caracteres disponíveis para cada nome.
2. Ataque: uma variável float para o atributo de ataque (ataque).
3. Defesa: uma variável float para o atributo de defesa (defesa).
4. Vida: uma variável float para o atributo de vida (vida).
5. Tipo: um vetor de caracteres para o tipo (tipo[10]), que podem ser: elétrico, água, fogo, gelo ou pedra. O tamanho 10 é o necessário para acomodar o maior nome dos tipos que possui 8 caracteres (elétrico), considerando também, o caractere nulo ('\0') e o caractere de nova linha ('\n').

Após a inclusão das bibliotecas e a definição da estrutura, foram declaradas as variáveis que foram utilizadas ao longo do programa. Para organizar o código de forma clara e organizada, o trabalho foi dividido em três outras funções além da função principal (main): "ler_arquivo", "vigoridade_da_luta" e "luta_pokemon". Cada uma dessas funções será detalhada posteriormente. Ao final da execução, o programa imprime para o usuário o resultado da batalha Pokémon.

2.1.1. Main

A função principal tem como objetivo controlar a execução do programa. Em primeiro lugar, são declaradas as variáveis que serão utilizadas ao longo do programa, sendo elas: o ponteiro "pokemon", do tipo POKEMON, o qual é inicializado como nulo e será utilizado para apontar para o vetor alocado dinamicamente durante a leitura e armazenamento dos dados. Além disso, os inteiros "N", "M" e "total" são declarados para armazenar, respectivamente, o número de Pokémon do primeiro treinador, o número de Pokémon do segundo treinador e a quantidade total de Pokémon envolvidos na batalha. É importante ressaltar que "N" e "M" foram usadas da mesma forma em que estavam no enunciado do trabalho prático.

Em seguida, a função "ler_arquivo" é chamada para realizar a leitura, armazenamento dos dados e impressão do arquivo .txt, conforme seu formato. Ademais, o código contém condicionais que o encerra caso erros sejam encontrados no processo de leitura que a fazem retornar "NULL", fazendo o programa ser encerrado instantaneamente.

Após a leitura, o valor de "total" é calculado; são declarados os inteiros "t1", para o treinador 1 e "t2" para o treinador 2, para indicar a posição atual de cada treinador no vetor "pokemon" e "turno" para facilitar o controle do turno de ataque durante a batalha.

A batalha é simulada por meio de um loop while-do, onde a função "luta_pokemon" é chamada repetidamente até que um treinador fique sem Pokémon com vida. Após cada confronto, é impresso o Pokémon vencedor da luta e quando algum treinador não possuir mais Pokémon, o loop se encerra.

Ao final da execução do loop, são impressas as seguintes informações: o treinador vencedor, os Pokémon que sobreviveram e os Pokémon que foram derrotados. Por fim, a memória alocada durante a leitura é liberada e o programa é encerrado, retornando '0'.

2.1.2. Leitura do arquivo

A função "ler_arquivo" é responsável por ler o arquivo de texto "pokemon1.txt", alocar dinamicamente espaço para armazenar os dados dos Pokémon contidos no arquivo e imprimir o conteúdo do arquivo exatamente como está formatado. A função também detecta e sinaliza os seguintes erros durante o processo de leitura:

- (i). Se o arquivo não puder ser encontrado ou aberto;
- (ii). Se o arquivo conter algum treinador com mais de 100 Pokémon;
- (iii). Se houver um erro durante a alocação dinâmica de memória.

Caso algum desses erros seja encontrado, a função retorna NULL, fazendo com que o programa seja encerrado na função main. Caso contrário, se nenhum erro for encontrado, a função retorna o vetor de "pokemon" alocado dinamicamente.

2.1.3. Vigoridade da luta

Cada Pokémon possui seu tipo, tendo ataques que são "super efetivos" (causam um dano maior) ou "não muito efetivos" (causam um dano menor) contra determinados tipos de Pokémon. Considerando isso, a função "vigoridade_da_luta" analisa os tipos dos Pokémon que vão se enfrentar na próxima luta. Para isso, ela compara as strings que representam os tipos dos Pokémon e retorna:

- (i). '1', caso o Pokémon do treinador 1 tenha vigoridade em relação ao Pokémon do treinador 2;
- (ii). '2', caso o Pokémon do treinador 2 tenha vigoridade em relação ao Pokémon do treinador 1;
- (iii). '0', caso nenhum Pokémon tenha vigoridade sobre o outro.

Observação: "Vigoridade" não é uma palavra existente na língua portuguesa, visto que é lexicalmente redundante. Entretanto, foi propositalmente criada desta forma para ser usada como um substantivo abstrato que denota a qualidade do que possui vigor ou a característica de quem tem energia física ou moral; força.

2.1.4. Simulação da luta

A função "luta_pokemon" simula a luta entre os Pokémon dos treinadores 1 e 2. Para isso, ela recebe como parâmetros o "turno" (o qual informa qual treinador é o próximo a atacar), "t1" e "t2" (indicando as posições dos Pokémon dos treinadores no vetor de "pokemon"), e um ponteiro para o vetor "pokemons". Em primeiro lugar, são declarados dois inteiros, "ataque_original_t1" e "ataque_original_t2", ambos para armazenar o poder de ataque original de cada Pokémon, de cada treinador antes de cada luta. Em seguida, é calculada a "vigoridade_da_luta", que determina a relação de força entre os tipos dos Pokémon envolvidos.

Feito isso, a luta é então simulada em um loop while-do, que continua enquanto os Pokémon estiverem com vida superior a 0. A simulação leva em conta o turno de ataque de cada treinador, somando um após o ataque do primeiro treinador e subtraindo um após o ataque do segundo treinador, mantendo assim os turnos alternados. Além disso, a função verifica se o Pokémon defensor ainda tem vida após cada ataque, mantém a vida do vencedor para a próxima luta e restaura o valor original de ataque dos Pokémon ao final de cada luta. Por fim, a função retorna o turno do próximo treinador a atacar.

2.2. Exemplo de Estilo texto esperado

O programa espera para a leitura do arquivo .txt, um texto com as seguintes informações:

- (i). O número de pokémon do treinador 1 e do treinador 2;
- (ii). Os 5 atributos de cada pokémon do treinador 1 e 2.

Exemplos:

a)

3 2

Squirtle 10 15 15 agua

Vulpix 15 15 15 fogo

Onix 5 20 20 pedra

Golem 20 5 10 pedra

Charmander 20 15 12 fogo

b)

5 5

Squirtle 15 25 40 agua

Sandshrew 17 13 28 pedra

Charmeleon 23 15 24 fogo

Raichu 19 16 28 eletrico

Cloyster 59 12 51 gelo

Charmander 25 12 36 fogo

Pikachu 25 15 24 eletrico

Dewgong 17 26 33 gelo

Wartortle 26 41 19 agua

Sandslash 62 31 40 pedra

c)

11 5

Pachirisu 15 17 9 eletrico

Ampharos 15 14 11 agua

Infernape 18 9 23 fogo

Spheal 14 18 18 gelo

Rampardos 9 18 22 pedra

Lotad 14 18 20 agua

Magmortar 21 16 24 fogo

Sealeo 15 19 28 gelo

Suicune 16 15 25 agua

Dugtrio 12 26 24 pedra

Charmander 18 12 26 fogo

Lombre 17 5 39 agua

Heatran 19 12 11 fogo

Walrein 15 14 20 gelo

Bonsly 11 19 17 pedra

Pikachu 35 30 24 eletrico

2.3. Exemplo de execução

Usando como exemplo, a entrada do exemplo a) do tópico 2.6:

Input:

```
3 2
Squirtle 10 15 15 agua
Vulpix 15 15 15 fogo
Onix 5 20 20 pedra
Golem 20 5 10 pedra
Charmander 20 15 12 fogo
```

Processo:

1. Leitura do número de Pokémon para cada treinador.
2. Alocação dinâmica com os dados dos Pokémons de cada treinador.

Vetor “pokemon” após leitura e alocação:

<i>Nome</i>	<i>Ataque</i>	<i>Defesa</i>	<i>Vida</i>	<i>Tipo</i>
Squirtle	10	15	15	agua
Vulpix	15	15	15	fogo
Onix	5	20	20	pedra
Golem	20	5	10	pedra
Charmander	20	15	12	fogo

Legenda: dada a lógica, os 3 primeiros são os do treinador 1 e os 2 seguintes, do treinador 2.

3. Impressão do conteúdo do arquivo texto.
4. Batalha (loop: vigoridade dos pokémons e luta dos Pokémons: Squirtle vs Golem, Squirtle vs Charmander).
5. Impressão do jogador vencedor.
6. Impressão dos pokémons sobreviventes e derrotados.
7. Liberação da memória alocada.
8. Retorno de valor '0';

Após simular a batalha se espera esta saída:

Output:

```
3 2
Squirtle 10 15 15 agua
Vulpix 15 15 15 fogo
Onix 5 20 20 pedra
Golem 20 5 10 pedra
Charmander 20 15 12 fogo

Squirtle venceu Golem
Charmander venceu Squirtle
Vulpix venceu Charmander
Jogador 1 venceu
Pokémon sobreviventes:
Vulpix
Onix
Pokémon derrotados:
Squirtle
Golem
Charmander
```

3. Testes e Erros

3.1. Processo de Criação do Algoritmo

O processo de criação do algoritmo foi uma experiência bastante interessante, visto que me permitiu aplicar vários conceitos abordados na disciplina. No entanto, cometi alguns erros ao longo do processo, principalmente erros lógicos. Utilizando os três primeiros exemplos de testes fornecidos, comparei os resultados esperados com os do programa e consegui identificar e corrigir esses erros.

3.2. Exemplos de Erros Comuns e Correções

Erro: não ter permissão para acessar o arquivo.

Correção: permitir o acesso ao arquivo.

Erro: declarar ataque defesa e vida como inteiros, sendo que ao averiguar a “vigoridade”, seus valores podem precisar de casas decimais.

Correção: declarar como float.

Erro: em “vigorosidade”, verificar o Pokémon do treinador 1 em relação ao do treinador 2, mas não o contrário.

Correção: realizar a comparação com strcmp também na direção oposta.

Erro: não controlar o turno de ataque, fazendo com o que, no decorrer, o treinador 1 tenha vantagem, já que ao derrotar um pokémon do treinador 2, ele permanecia atacando.

Correção: criar a variável “turno”, inicializada com o valor ‘1’ (indicando o turno do treinador 1). Após o ataque do treinador 1, é somado 1 ao “turno” (1+1=2), passando o turno para o treinador 2. Após o ataque do treinador 2, é subtraído 1 de “turno” (2-1=1), voltando ao turno do treinador 1. Esse processo é repetido até que a luta se encerre. É importante notar que o valor de “turno” é retornado após cada luta pela função “luta_pokemon” e indica qual treinador será o próximo a atacar.

3.3. Testes Realizados

Para garantir o funcionamento correto, o programa foi testado com todos os cinco casos de testes disponibilizados.

Caso de Teste 1		
Entrada	Saída esperada	Saída do programa
3 2 Squirtle 10 15 15 agua Vulpix 15 15 15 fogo Onix 5 20 20 pedra Golem 20 5 10 pedra Charmander 20 15 12 fogo	.txt Squirtle venceu Golem Charmander venceu Squirtle Vulpix venceu Charmander Jogador 1 venceu Pokemon sobreviventes: Vulpix Onix Pokemon derrotados: Squirtle Golem Charmander	3 2 Squirtle 10 15 15 agua Vulpix 15 15 15 fogo Onix 5 20 20 pedra Golem 20 5 10 pedra Charmander 20 15 12 fogo Squirtle venceu Golem Charmander venceu Squirtle Vulpix venceu Charmander Jogador 1 venceu Pokemon sobreviventes: Vulpix Onix Pokemon derrotados: Squirtle Golem Charmander

Caso de Teste 2		
Entrada	Saída esperada	Saída do programa
5 5 Squirtle 15 25 40 agua Sandshrew 17 13 28 pedra Charmeleon 23 15 24 fogo Raichu 19 16 28 eletrico Cloyster 59 12 51 gelo Charmander 25 12 36 fogo Pikachu 25 15 24 eletrico Dewgong 17 26 33 gelo Wartortle 26 41 19 agua Sandslash 62 31 40 pedra	.txt Squirtle venceu Charmander Pikachu venceu Squirtle Sandshrew venceu Pikachu Dewgong venceu Sandshrew Charmeleon venceu Dewgong Wartortle venceu Charmeleon Wartortle venceu Raichu Cloyster venceu Wartortle Sandslash venceu Cloyster Jogador 2 venceu Pokémons sobreviventes: Sandslash Pokémons derrotados: Squirtle Sandshrew Charmeleon Raichu Cloyster Charmander Pikachu Dewgong Wartortle	5 5 Squirtle 15 25 40 agua Sandshrew 17 13 28 pedra Charmeleon 23 15 24 fogo Raichu 19 16 28 eletrico Cloyster 59 12 51 gelo Charmander 25 12 36 fogo Pikachu 25 15 24 eletrico Dewgong 17 26 33 gelo Wartortle 26 41 19 agua Sandslash 62 31 40 pedra Squirtle venceu Charmander Pikachu venceu Squirtle Sandshrew venceu Pikachu Dewgong venceu Sandshrew Charmeleon venceu Dewgong Wartortle venceu Charmeleon Wartortle venceu Raichu Cloyster venceu Wartortle Sandslash venceu Cloyster Jogador 2 venceu Pokemon sobreviventes: Sandslash Pokemon derrotados: Squirtle Sandshrew Charmeleon Raichu Cloyster Charmander Pikachu Dewgong Wartortle

Caso de Teste 3		
Entrada	Saída esperada	Saída do programa
11 5 Pachirisu 15 17 9 eletrico Ampharos 15 14 11 agua Infernape 18 9 23 fogo	.txt Pachirisu venceu Lombre Heatran venceu Pachirisu Ampharos venceu Heatran	11 5 Pachirisu 15 17 9 eletrico Ampharos 15 14 11 agua Infernape 18 9 23 fogo

<p> Spheal 14 18 18 gelo Rampardos 9 18 22 pedra Lotad 14 18 20 agua Magmortar 21 16 24 fogo Sealeo 15 19 28 gelo Suicune 16 15 25 agua Dugtrio 12 26 24 pedra Charmander 18 12 26 fogo Lombre 17 5 39 agua Heatran 19 12 11 fogo Walrein 15 14 20 gelo Bonsly 11 19 17 pedra Pikachu 35 30 24 eletrico </p>	<p> Walrein venceu Ampharos Infernape venceu Walrein Bonsly venceu Infernape Spheal venceu Bonsly Pikachu venceu Spheal Pikachu venceu Rampardos Pikachu venceu Lotad Pikachu venceu Magmortar Pikachu venceu Sealeo Pikachu venceu Suicune Pikachu venceu Dugtrio Pikachu venceu Charmander Jogador 2 venceu Pokémons sobreviventes: Pikachu Pokémons derrotados: Pachirisu Ampharos Infernape Spheal Rampardos Lotad Magmortar Sealeo Suicune Dugtrio Charmander Lombre Heatran Walrein Bonsly </p>	<p> Spheal 14 18 18 gelo Rampardos 9 18 22 pedra Lotad 14 18 20 agua Magmortar 21 16 24 fogo Sealeo 15 19 28 gelo Suicune 16 15 25 agua Dugtrio 12 26 24 pedra Charmander 18 12 26 fogo Lombre 17 5 39 agua Heatran 19 12 11 fogo Walrein 15 14 20 gelo Bonsly 11 19 17 pedra Pikachu 35 30 24 eletrico </p> <p> Pachirisu venceu Lombre Heatran venceu Pachirisu Ampharos venceu Heatran Walrein venceu Ampharos Infernape venceu Walrein Bonsly venceu Infernape Spheal venceu Bonsly Pikachu venceu Spheal Pikachu venceu Rampardos Pikachu venceu Lotad Pikachu venceu Magmortar Pikachu venceu Sealeo Pikachu venceu Suicune Pikachu venceu Dugtrio Pikachu venceu Charmander Jogador 2 venceu Pokemon sobreviventes: Pikachu Pokemon derrotados: Pachirisu Ampharos Infernape Spheal Rampardos Lotad Magmortar Sealeo Suicune Dugtrio Charmander Lombre Heatran Walrein Bonsly </p>
--	---	--

Caso de Teste 4		
Entrada	Saída esperada	Saída do programa
26 26 Squirtle 10 15 15 agua Bastiodon 18 30 38 pedra Shinx 13 15 28 eletrico Chimchar 12 14 9 fogo Cloyster 20 25 35 gelo Blastoise 15 20 25 agua Onix 18 25 35 pedra Pikachu 15 10 30 eletrico Remoraid 12 14 18 agua Spheal 15 18 22 gelo Suicune 25 22 40 agua Rampardos 25 20 40 pedra Lombre 15 14 20 agua Charmander 20 15 25 fogo Ninetales 25 20 35 fogo Wartortle 15 22 28 agua Chimchar 20 16 25 fogo Magnetron 20 18 32 eletrico Dugtrio 15 22 30 pedra Magmortar 25 20 32 fogo Sealeo 20 25 30 gelo Bonsly 12 18 25 pedra Raichu 18 12 35 eletrico Lapras 25 30 40 gelo Luxio 16 18 30 eletrico Piloswine 18 22 28 gelo Snorunt 16 18 25 gelo Octillery 20 18 28 agua Glalie 22 20 30 gelo Heatran 28 24 38 fogo Diglett 12 20 28 pedra Psyduck 14 16 22 agua Pachirisu 15 12 30 eletrico Lotad 10 12 15 agua Cranidos 20 18 32 pedra Vulpix 18 15 22 fogo Golduck 18 20 30 agua Wartortle 12 18 20 agua Growlithe 11 13 14 fogo Dewgong 18 22 30 gelo Sheldon 15 25 30 pedra Magnezone 12 20 28 eletrico Monferno 22 18 28 fogo Ninetales 19 17 27 fogo Vulcorb 14 16 25 eletrico Charmeleon 22 18 30 fogo Swinub 12 14 20 gelo Luxray 22 20 35 eletrico Infernape 30 22 35 fogo	.txt Snorunt venceu Squirtle Bastiodon venceu Snorunt Bastiodon venceu Octillery Glalie venceu Bastiodon Glalie venceu Shinx Glalie venceu Chimchar Cloyster venceu Glalie Heatran venceu Cloyster Heatran venceu Blastoise Heatran venceu Onix Heatran venceu Pikachu Heatran venceu Remoraid Heatran venceu Spheal Suicune venceu Heatran Suicune venceu Diglett Suicune venceu Psyduck Suicune venceu Pachirisu Suicune venceu Lotad Suicune venceu Cranidos Suicune venceu Vulpix Suicune venceu Golduck Suicune venceu Wartortle Suicune venceu Growlithe Dewgong venceu Suicune Rampardos venceu Dewgong Sheldon venceu Rampardos Lombre venceu Sheldon Lombre venceu Magnemite Monferno venceu Lombre Monferno venceu Charmander Ninetales venceu Monferno Ninetales venceu Ninetales Ninetales venceu Vulcorb Ninetales venceu Charmeleon Ninetales venceu Swinub Luxray venceu Ninetales Luxray venceu Wartortle Chimchar venceu Luxray Infernape venceu Chimchar Infernape venceu Magnetron Infernape venceu Dugtrio Infernape venceu Magmortar Infernape venceu Sealeo Infernape venceu Bonsly Infernape venceu Raichu Infernape venceu Lapras Luxio venceu Infernape Magnezone venceu Luxio Magnezone venceu Piloswine	26 26 Squirtle 10 15 15 agua Bastiodon 18 30 38 pedra Shinx 13 15 28 eletrico Chimchar 12 14 9 fogo Cloyster 20 25 35 gelo Blastoise 15 20 25 agua Onix 18 25 35 pedra Pikachu 15 10 30 eletrico Remoraid 12 14 18 agua Spheal 15 18 22 gelo Suicune 25 22 40 agua Rampardos 25 20 40 pedra Lombre 15 14 20 agua Charmander 20 15 25 fogo Ninetales 25 20 35 fogo Wartortle 15 22 28 agua Chimchar 20 16 25 fogo Magnetron 20 18 32 eletrico Dugtrio 15 22 30 pedra Magmortar 25 20 32 fogo Sealeo 20 25 30 gelo Bonsly 12 18 25 pedra Raichu 18 12 35 eletrico Lapras 25 30 40 gelo Luxio 16 18 30 eletrico Piloswine 18 22 28 gelo Snorunt 16 18 25 gelo Octillery 20 18 28 agua Glalie 22 20 30 gelo Heatran 28 24 38 fogo Diglett 12 20 28 pedra Psyduck 14 16 22 agua Pachirisu 15 12 30 eletrico Lotad 10 12 15 agua Cranidos 20 18 32 pedra Vulpix 18 15 22 fogo Golduck 18 20 30 agua Wartortle 12 18 20 agua Growlithe 11 13 14 fogo Dewgong 18 22 30 gelo Sheldon 15 25 30 pedra Magnezone 12 20 28 eletrico Monferno 22 18 28 fogo Ninetales 19 17 27 fogo Vulcorb 14 16 25 eletrico Charmeleon 22 18 30 fogo Swinub 12 14 20 gelo Luxray 22 20 35 eletrico Infernape 30 22 35 fogo

Magnezone 25 22 40 eletrico Growlithe 15 12 18 fogo Blastoise 20 25 40 agua	Jogador 2 venceu Pokémons sobreviventes: Magnezone Growlithe Blastoise Pokémons derrotados: Squirtle Bastiodon Shinx Chimchar Cloyster Blastoise Onix Pikachu Remoraid Speal Suicune Rampardos Lombre Charmander Ninetales Wartortle Chimchar Magneton Dugtrio Magmortar Sealeo Bonsly Raichu Lapras Luxio Piloswine Snorunt Octillery Glalie Heatran Diglett Psyduck Pachirisu Lotad Cranidos Vulpix Golduck Wartortle Growlithe Dewgong Sheldon Magnemite Monferno Ninetales Voltorb Charmeleon Swinub Luxray	Magnezone 25 22 40 eletrico Growlithe 15 12 18 fogo Blastoise 20 25 40 agua Snorunt venceu Squirtle Bastiodon venceu Snorunt Bastiodon venceu Octillery Glalie venceu Bastiodon Glalie venceu Shinx Glalie venceu Chimchar Cloyster venceu Glalie Heatran venceu Cloyster Heatran venceu Blastoise Heatran venceu Onix Heatran venceu Pikachu Heatran venceu Remoraid Heatran venceu Speal Suicune venceu Heatran Suicune venceu Diglett Suicune venceu Psyduck Suicune venceu Pachirisu Suicune venceu Lotad Suicune venceu Cranidos Suicune venceu Vulpix Suicune venceu Golduck Suicune venceu Wartortle Suicune venceu Growlithe Dewgong venceu Suicune Rampardos venceu Dewgong Sheldon venceu Rampardos Lombre venceu Sheldon Lombre venceu Magnemite Monferno venceu Lombre Monferno venceu Charmander Ninetales venceu Monferno Ninetales venceu Ninetales Ninetales venceu Voltorb Ninetales venceu Charmeleon Ninetales venceu Swinub Luxray venceu Ninetales Luxray venceu Wartortle Chimchar venceu Luxray Infernape venceu Chimchar Infernape venceu Magnezone Infernape venceu Dugtrio Infernape venceu Magmortar Infernape venceu Sealeo Infernape venceu Bonsly Infernape venceu Raichu Infernape venceu Lapras Luxio venceu Infernape Magnezone venceu Luxio Magnezone venceu Piloswine Jogador 2 venceu
---	--	---

	Infernape	<p>Pokemon sobreviventes:</p> <p>Magnezone</p> <p>Growlithe</p> <p>Blastoise</p> <p>Pokemon derrotados:</p> <p>Squirtle</p> <p>Bastiodon</p> <p>Shinx</p> <p>Chimchar</p> <p>Cloyster</p> <p>Blastoise</p> <p>Onix</p> <p>Pikachu</p> <p>Remoraid</p> <p>Spheal</p> <p>Suicune</p> <p>Rampardos</p> <p>Lombre</p> <p>Charmander</p> <p>Ninetales</p> <p>Wartortle</p> <p>Chimchar</p> <p>Magneton</p> <p>Dugtrio</p> <p>Magmortar</p> <p>Sealeo</p> <p>Bonsly</p> <p>Raichu</p> <p>Lapras</p> <p>Luxio</p> <p>Piloswine</p> <p>Snorunt</p> <p>Octillery</p> <p>Glalie</p> <p>Heatran</p> <p>Diglett</p> <p>Psyduck</p> <p>Pachirisu</p> <p>Lotad</p> <p>Cranidos</p> <p>Vulpix</p> <p>Golduck</p> <p>Wartortle</p> <p>Growlithe</p> <p>Dewgong</p> <p>Sheldon</p> <p>Magnemite</p> <p>Monferno</p> <p>Ninetales</p> <p>Voltorb</p> <p>Charmeleon</p> <p>Swinub</p> <p>Luxray</p> <p>Infernape</p>
--	-----------	---

Caso de Teste 5

Entrada	Saída esperada	Saída do programa
<p>5 7</p> <p>Golduck 11 18 13 agua</p> <p>Geodude 11 23 22 pedra</p> <p>Raichu 14 18 21 eletrico</p> <p>Rapidash 24 11 24 fogo</p> <p>Rhydon 15 11 21 pedra</p> <p>Charizard 20 22 19 fogo</p> <p>Slowbro 21 16 25 agua</p> <p>Magnemite 18 22 20 eletrico</p> <p>Dewgong 22 14 22 agua</p> <p>Tentacool 14 11 24 agua</p> <p>Graveler 17 10 19 pedra</p> <p>Marowak 19 22 21 pedra</p>	<p>.txt</p> <p>Charizard venceu Golduck</p> <p>Geodude venceu Charizard</p> <p>Slowbro venceu Geodude</p> <p>Raichu venceu Slowbro</p> <p>Magnemite venceu Raichu</p> <p>Magnemite venceu Rapidash</p> <p>Rhydon venceu Magnemite</p> <p>Dewgong venceu Rhydon</p> <p>Jogador 2 venceu</p> <p>Pokémons sobreviventes:</p> <p>Dewgong</p> <p>Tentacool</p> <p>Graveler</p> <p>Marowak</p> <p>Pokémons derrotados:</p> <p>Golduck</p> <p>Geodude</p> <p>Raichu</p> <p>Rapidash</p> <p>Rhydon</p> <p>Charizard</p> <p>Slowbro</p> <p>Magnemite</p>	<p>5 7</p> <p>Golduck 11 18 13 agua</p> <p>Geodude 11 23 22 pedra</p> <p>Raichu 14 18 21 eletrico</p> <p>Rapidash 24 11 24 fogo</p> <p>Rhydon 15 11 21 pedra</p> <p>Charizard 20 22 19 fogo</p> <p>Slowbro 21 16 25 agua</p> <p>Magnemite 18 22 20 eletrico</p> <p>Dewgong 22 14 22 agua</p> <p>Tentacool 14 11 24 agua</p> <p>Graveler 17 10 19 pedra</p> <p>Marowak 19 22 21 pedra</p> <p>Charizard venceu Golduck</p> <p>Geodude venceu Charizard</p> <p>Slowbro venceu Geodude</p> <p>Raichu venceu Slowbro</p> <p>Magnemite venceu Raichu</p> <p>Magnemite venceu Rapidash</p> <p>Rhydon venceu Magnemite</p> <p>Dewgong venceu Rhydon</p> <p>Jogador 2 venceu</p> <p>Pokemon sobreviventes:</p> <p>Dewgong</p> <p>Tentacool</p> <p>Graveler</p> <p>Marowak</p> <p>Pokemon derrotados:</p> <p>Golduck</p> <p>Geodude</p> <p>Raichu</p> <p>Rapidash</p> <p>Rhydon</p> <p>Charizard</p> <p>Slowbro</p> <p>Magnemite</p>

4. Conclusão

O trabalho prático de Programação e Desenvolvimento de Software I foi fundamental para aprimorar habilidades práticas de programação, visto que englobou quase todos os conceitos discutidos em sala de aula e proporcionou uma aplicação prática em um ambiente criativo. A escolha do tema, baseado no universo Pokémon, foi especialmente relevante devido à sua popularidade, facilitando a interpretação e o progresso da criação do programa. Além disso, o suporte contínuo do professor Clodoveu e do monitor Estanislau Filho foi crucial para o esclarecimento de dúvidas e para a conclusão do trabalho. Por fim, para tornar esse programa do projeto mais interessante, seria proveitoso os temas serem alterados ao longo do tempo, mas mantendo-se sempre o nível tanto de complexidade, quanto de criatividade ao programar.