Descriptografando a profecia de Link
Trabalho Prático - Projeto de Análise de Algoritmos

Jhonata Miranda-3859, Vinícius Mendes-3881, Maurício Okuyama-4239

[1]Universidade Federal de Viçosa - Campus Florestal (UFVCaf)

jhonata.miranda@ufv.br, vinicius.o.mendes@ufv.br, mauricio.okuyama@ufv.br

1. Introdução

O trabalho em questão tem como objetivo abordar os conteúdos trabalhados na disciplina de Projeto e Análise de Algoritmos, em específico, como visto nas aulas teóricas. A priori, é válido dizer que o processo de descriptografia baseia-se em uma série de passos que devem ser levados em conta ao tratar do texto repassado.

Diante disso, é observado a exigência da utilização de tabelas de frequência para buscar soluções que satisfazem a necessidade da princesa Zelda, dentre elas, uma tabela de frequência relativa das letras na língua portuguesa juntamente com a obtida no próprio texto. Dessa forma, foi possível traçar um paralelo entre ambas, e assim encontrar formas de associar os dois alfabetos obtidos solucionando o problema da princesa.

2. Desenvolvimento

2.1. O problema e a solução

O trabalho consiste na descriptografia de um texto utilizando uma chave desconhecida. Com métodos de análise de frequência, encontrando padrões, tanto no texto criptografado ou parcialmente descriptografado e mapeamento parcial da chave, deve-se interagir com o programa para que seja feita a transcrição do texto para o idioma Português.

Contextualizando, o herói recebe uma profecia para restaurar o equilíbrio de seu mundo e pede a ajuda dos alunos de Ciência da Computação. Utilizamos uma criptoanálise clássica para decifrar a profecia, criptoanálise essa que consiste na implementação de algoritmos como o Shift-And exato e o Shift-And aproximado. Para descobrir a frequência das letras no texto, utilizamos o método de contagem simples de letras e exibindo-as juntamente com a frequência das letras em Português.

2.2. Compilação, execução

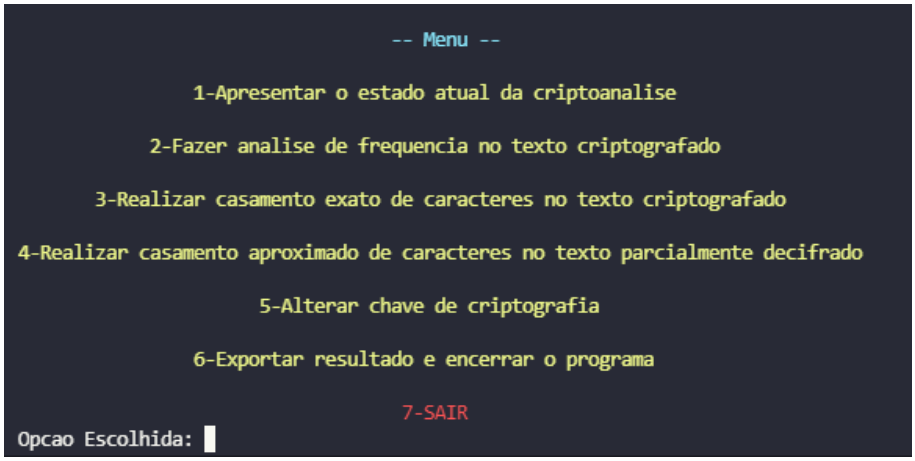
Para a compilação dos arquivos-fonte, deve se usar o comando **make run** para compilar e executar o makefile que está dentro do caminho origem contido na pasta do trabalho. Após isso, será exibido um menu requerindo o nome do arquivo a ser lido pelo algoritmo. É válido ressaltar que para execução correta do projeto, o arquivo a ser lido deve ser inserido na pasta **data**, contida dentro do arquivo de desenvolvimento e o caminho a ser passado deve seguir o exemplo: **data/nomedoarquivo.txt**.

Caso seja necessário limpar o executável e os arquivos de compilação, pode-se utilizar o comando **make clean**.

```
# compilar e executar
make
./build/main

# compilar e executar em um comando
make run

# limpar executável e arquivos de compilação
make clean
```

Comandos para utilização do Makefile

Logo após, será exibido um menu solicitando a opção ao usuário. Inicialmente texto parcialmente descriptografado será igual ao texto original, sendo necessário alterar a chave de criptografia para que ele altere. Será exibido em verde as alterações feitas no texto parcialmente criptografado.



Menu

2.3. Estruturas

São utilizados duas estruturas neste trabalho. A estrutura Frequencia armazena uma letra a a quantidade de vezes que essa letra aparece. Já a estrutura Texto armazena o texto criptografado, o texto parcialmente descriptografado, a chave e a lista de frequência das letras.

```c
typedef struct Frequencia
{
    char letra;
    int qtd;
}frequencia;

typedef struct Texto
{
    char *criptografado;
    char *parcial;
    char chave[TAMANHO_ALFABETO+1];
    frequencia lista_frequencia[TAMANHO_ALFABETO];
} texto;
```

Estruturas Frequencia e Texto

2.4. Decisões importantes

Na busca no texto criptografado, dadas as opções o grupo decidiu utilizar o algoritmo Shift-And exato. Esta decisão se justifica pelo fato de utilizarmos uma lógica semelhante ao realizarmos busca no texto parcialmente criptografado, sendo neste ultimo implementado o Shift-And aproximado. Tanto neste algoritmo quanto no Shift-And aproximado, não é aceito o uso de caracteres que não compõem o alfabeto Português.

Nos contextos de busca, foi decidido utilizar a função *toupper* da biblioteca **ctype.h**, que é responsável por adaptar o padrão repassado pelo usuário, transformando-o sempre em letras maiúsculas. Dessa maneira, o processo de busca é facilitado, visto que, reduz a probabilidade de erro proveniente do utilizador do algoritmo.

Para mostrar as mudanças feitas durante o processo de alteração da chave de criptografia, o grupo optou por exibir o estado atual da criptoanálise em uma cor diferente da cor padrão do texto.

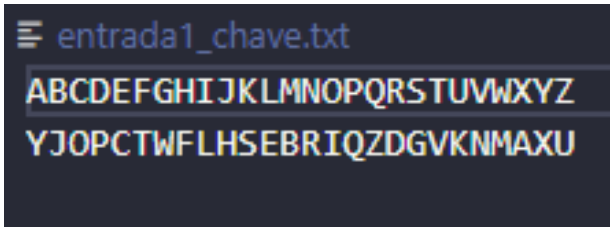
As letras alteradas são exibidas em verde, e assim que o processo de descriptografia termina, todos os caracteres mapeados são representados pela cor.

```
=== Chave ===
ABCDEFGHIJKLMNOPQRSTUVWXYZ
Y   C         Q
=== Texto parcialmente decifrado ===
ELG ZKE A OAEABLPAPE GE AJAVE BALG KBA NEU GIJDE FXDKEE. VARVAG NEUEG A NLPA REGVA VEDDA TIL ABEAOAPA, VARVAG NEUEG I DEL BAEETLOI GE IPIG A WDAOA PAG
 VDEG PEKGAG. PID NEUEG I PINI NLNEK AOLBA PIG OEKG, TKWLRPI PAG GIBJDAG PA GKPEDTLOLE, PID NEUEG A VEDDA GE ATIWIK EB PLEKNLI, E PID IKVDAG VARVAG NE
UEG A DEAELPAPE GE PLGVIDOEK ERVDE PEARIG. E AWIDA EGVA VAENEU GEHA A JAVAEFA TLRAE. VIPAG AG ELRFAG PI VEBPI GE OIELPLDAI, E VIPIG IG ADZKLRLBLWIG, K
B PLA PEDDIVAPIG, DEVIDRADAI.

PIDEB, ZKARPI TIL ZKE I DELRI EGVENE PEGABPADAPI? ALRPA ZKE VADPLI, I FEDIL PI VEBPI GEBPDE GKDWE ZKARPI FXDKEE EGVA EB PEDLWI. EB RIBE PE OAPA ADNIDE
 E OAPA FAJLVARVE PA TEIDEGVA, EB RIBE PE OAPA DLI, BAD E EAWI PINIAPI PEEIG UIDAG, EB RIBE PE OAPA BIRVARFA WKADPAPA PEEIG WIDIRG, E PID VIPAG AG IKV
DAG ODLAVKDAG ZKE OIEALGVEB EB FADBIRLA, PEGPE IG VMLEL AVE AG TAPAG. PID VIPIG EGGEG, I FEDIL PI VEBPI GEBPDE EKVIK E GALK NLVIDLIGI.

PADA OIBPEEVAD GKA PDIALBA BLGGAI, ELRS, EGVEHA AVERVI. OIRVDA VIPAG ABEAOAG GAI EALWLPIG VIPIG IG DEOKDGIG. ZKARPI IG AGVDIG GE AELRFADEB, KBA OIRNED
WEROLA EZKLNAEERVE PENEDA GE EDWKED RA VEDDA: ZKE IG GEVE GAJLIG EGVEHAB EB GEKG PIGVIG OLDOKRGODLVIG; ZKE IG ZKAVDI WLWARVEG GEHAB AOIDPAPIG E OIRNIO
APIG; ZKE IG ZKAVDI EGPLDLVIG PA EKU GE AELRFEB AIG WLWARVEG EB OAPA PIRVI OADPEAE; E ZKE AG PEPDAG PAG VDEG PEKGAG EGVEHAB ELWAPAG RI OERVDI PE VKPI.
 GI AGGLB VIPI I PIVEROLAE PE UEEPA GEDA PEGPEDVAPI E I PEGVLRI EGVADA GEEAPI.
```

Exemplo de alterações na chave de criptografia

2.5. Testes e Resultados

Para realizar testes no programa utilizamos dois arquivos de texto. Um com a frase exemplificada na documentação e outro arquivo mais longo enviado por e-mail pelo monitor. Executando a análise de frequência e descriptografando os textos, obtivemos os seguintes resultados:



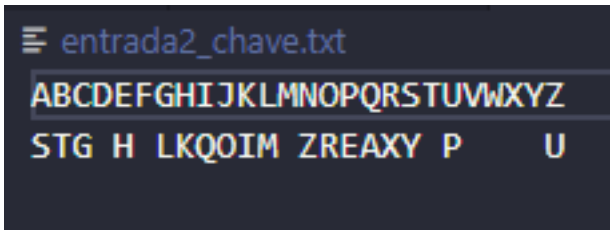
Chave após descriptografia do texto 1

EIS QUE A CALAMIDADE SE ABATE MAIS UMA VEZ SOBRE HARULE. TANTAS VEZES A VIDA NESTA TERRA FOI AMEACADA, TANTAS VEZES O REI MALEFICO SE OPOS A GRACA DAS TRES DEUSAS. POR VEZES O POVO VIVEU ACIMA DOS CEUS, FUGINDO DAS SOMBRAS DA SUPERFICIE, POR VEZES A TERRA SE AFOGOU EM DILUVIO, E POR OUTRAS TANTAS VEZES A REALIDADE SE DISTORCEU ENTRE PLANOS. E AGORA ESTA TALVEZ SEJA A BATALHA FINAL. TODAS AS LINHAS DO TEMPO SE COLIDIRAO, E TODOS OS ARQUINIMIGOS, UM DIA DERROTADOS, RETORNARAO.

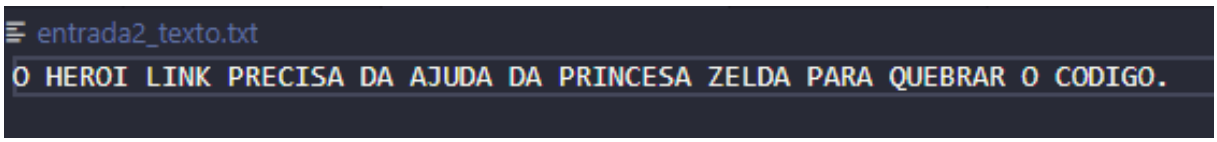
POREM, QUANDO FOI QUE O REINO ESTEVE DESAMPARADO? AINDA QUE TARDIO, O HEROI DO TEMPO SEMPRE SURGE QUANDO HARULE ESTA EM PERIGO. EM NOME DE CADA ARVORE E CADA HABITANTE DA FLORESTA, EM NOME DE CADA RIO, MAR E LAGO POVOADO PELOS ZORAS, EM NOME DE CADA MONTANHA GUARDADA PELOS GORONS, E POR TODAS AS OUTRAS CRIATURAS QUE COEXISTEM EM HARMONIA, DESDE OS TWILI ATE AS FADAS. POR TODOS ESSES, O HEROI DO TEMPO SEMPRE LUTOU E SAIU VITORIOSO.

PARA COMPLETAR SUA PROXIMA MISSAO, LINK, ESTEJA ATENTO. CONTRA TODAS AMEACAS SAO EXIGIDOS TODOS OS RECURSOS. QUANDO OS ASTROS SE ALINHAREM, UMA CONVERGENCIA EQUIVALENTE DEVERA SE ERGUER NA TERRA: QUE OS SETE SABIOS ESTEJAM EM SEUS POSTOS CIRCUNSCRITOS; QUE OS QUATRO GIGANTES SEJAM ACORDADOS E CONVOCADOS; QUE OS QUATRO ESPIRITOS DA LUZ SE ALINHEM AOS GIGANTES EM CADA PONTO CARDEAL; E QUE AS PEDRAS DAS TRES DEUSAS ESTEJAM LIGADAS NO CENTRO DE TUDO. SO ASSIM TODO O POTENCIAL DE ZELDA SERA DESPERTADO E O DESTINO ESTARA SELADO.

Texto 1 descriptografado



Chave após descriptografia do texto 2



Texto 2 descriptografado

3. Conclusão

Conclui-se que os algoritmos foram implementados de acordo com o que foi solicitado na documentação e que o grupo desempenhou bem as atividades propostas. É de suma importância ressaltar que os materiais disponibilizados em aula, juntamente com os websites referenciados ajudaram a concluir o projeto.

A utilização da análise de frequência juntamente com a busca por padrões nos textos, colaborou com a descriptografia, servindo como um material de consulta ao tentar supor as letras que foram mapeadas na tentativa de construir uma chave que faça a descriptografia correta do texto.

4. Referências

Frequência de letras na língua portuguesa: https://pt.wikipedia.org/wiki/Frequłncia_de_letras

Formas de colorir a saída no terminal: https://stackoverflow.com/questions/3219393/stdlib-and-colored-output-in-c

Algoritmo Quick Sort para ordenação da frequência de letras: https://devdocs.io/c/algorithm/qsort

Algoritmos ShiftAnd e ShiftAndAproximado do professor Ziviani: http://www2.dcc.ufmg.br/livros/algoritmos/cap8/codigo/c/8.1a8.6e8.8-pesquisacadeia.c