Trabalho de Programação 3

Processador Intel (80x86)

1. Descrição Geral

Neste trabalho você deverá desenvolver um programa escrito em assembler (linguagem de montagem) do 8086, capaz de ler um arquivo (arquivo de entrada) com informações sobre as base nitrogenadas de uma fita de DNA (Ácido Desoxirribonucleico). As bases nitrogenadas serão identificadas pelas letras “ATCG”, correspondentes às bases Adenina, Timina, Citosina e Guanina.

O nome do arquivo de entrada será fornecido na linha de comando assim como as opções que controlam os cálculos a serem realizados sobre o arquivo lido.

O arquivo de entrada pode ter qualquer nome válido no MS-DOS. Além disso, o arquivo deverá ser tratado como uma sequência de caracteres que representação as bases nitrogenadas.

Como resultado, seu programa deverá escrever o resultado dos cálculos solicitados em um arquivo de saída. Além disso, seu programa deverá escrever na tela um resumo das informações processadas.

Seu programa deverá ser desenvolvido em linguagem simbólica de montagem do processador 8086 da Intel e executado no ambiente DosBox. Para montagem de seu programa fonte será usado o montador MASM 6.11.

2. Chamada do programa

Seu programa será chamado através da “linha de comando”. No final deste documento você encontrará a forma como obter um string com os caracteres digitados na linha de comando. Esse string contém todas as informações digitadas na linha de comando, exceto o nome de seu programa.

Seu programa deve processar o string fornecido na linha de comando, onde irá encontrar uma série de opções que vão determinar o que seu programa deverá realizar. As opções são letras MINÚSCULAS antecedidas por um sinal de “-“ (menos), e poderão ser encontradas na linha de comando em qualquer ordem. Observe que algumas opções possuem parâmetros, como é o caso da opção “-f”.

As opções que seu programa deve ser capaz de processar são as seguintes:

- “-f”, seguida do nome do arquivo de entrada;
- “-o”, seguida do nome do arquivo de saída;
- “-n”, seguida de um número inteiro sem sinal;
- “-atcg+”

A opção “-f” deve ser seguida pelo nome do arquivo de entrada. Os detalhes do arquivo de entrada estão na sessão 3 a seguir.

A opção “-o” deve ser seguida pelo nome do arquivo de saída dos resultados. Se essa opção não for fornecida, o programa deverá escrever os resultados em um arquivo chamado de “a.out”.

A opção “-n” deve ser seguida por um número inteiro sem sinal. Esse número será utilizado no processamento do arquivo. Ver a sessão 4 para detalhes desse processamento.

A opção “-atcg+” informa as informações a serem colocadas no arquivo de saída. Cada símbolo, que pode aparecer em qualquer ordem, tem o seguinte significado:

- “a”, para colocar no arquivo de saída as informações sobre a base nitrogenada “A”;
- “t” , para colocar no arquivo de saída as informações sobre a base nitrogenada “T”;
- “c”, para colocar no arquivo de saída as informações sobre a base nitrogenada “C”;
- “g”, para colocar no arquivo de saída as informações sobre a base nitrogenada “G”;

- "+", para colocar no arquivo de saída as informações acumuladas das bases "A" e "T" e também das bases "C" e "G".

Essa opção pode ser formada pela combinação de qualquer dos cinco símbolos previstos. Por exemplo, se a opção for "-ct", então deverão ser colocadas no arquivo de saída as informações sobre as base "C" (Citosina) e "T" (Timina).

Observar que os símbolos podem estar em qualquer ordem. Por exemplo, "-a+c" é o mesmo que "-+ca".

São exemplos de linhas de comando, onde "prog" é o nome do programa:

- prog -f arq.txt -n 5 -atcg
  - Processa o arquivo de entrada "arq.txt", usando grupos com 5 bases, e gera como resultado os contadores de cada uma das bases nitrogenadas, que serão colocados no arquivo "a.out".
- prog -o file.txt -f input.txt –n 10 -+
  - Processo o arquivo de entrada "input.txt", usando grupos com 10 bases, e gera como resultado o somatório das base "A+T" e "C+G", que serão colocados no arquivo de saída "file.txt".

3. Formato do arquivo de entrada

O arquivo de entrada contém um string de letras que representam as bases nitrogenadas que formam a fita de DNA a ser processada. Esse string de letras pode estar separado por bytes CR e/ou LF, que devem ser ignorados durante o processamento dos dados (esses caracteres são usados apenas para facilitar a geração dos arquivos de entrada).

O arquivo terá, no máximo, 10.000 bases (letras).

As letras que formam o string no arquivo de entrada são, exclusivamente, "A", "T", "C" e "G". Qualquer outra letra ou símbolo é inválido.

São consideradas situações de erro relacionadas com o arquivo de entrada e que devem ser informadas na tela:

- Arquivo de entrada não existe. Nesse caso, deve ser informado o nome do arquivo de entrada;
- Quantidade de letras no arquivo de entrada é muito pequeno para ser processado. Nesse caso, deve ser informado qual é a quantidade mínima de letras que devem existir no arquivo de entrada;
- As opções informadas na linha de comando não são suficientes para processar os dados do arquivo de entrada. Neste caso, deve ser informado quais são as informações que estão faltando;
- Alguma opção existente na linha de comando é inválida. Nesse caso, seu programa deve informar na tela o erro e qual é a opção inválida;
- Alguma opção existente na linha de comando contém parâmetros inválidos. Neste caso, seu programa deve informar na tela o erro, a opção e o parâmetro inválidos;
- Arquivo muito grande. Nesse caso, deve ser informado que o arquivo possui mais de 10.000 bases nitrogenadas;
- Foi encontrado no arquivo de entrada uma letra inválida. Nesse caso, deve ser informada a letra errada e a linha do arquivo onde essa letra foi encontrada.

No caso da identificação de erro, o programa devem informar a situação identificada e encerrar sua execução.

4. Processamento do arquivo de entrada

Após ter lido as opções fornecidas na linha de comando, seu programa deverá ler o arquivo de entrada e realizar o processamento necessário para gerar os resultados solicitados.

Seu programa deverá contar as letras A,T,C e G, que representam as bases nitrogenadas, existentes em grupos com "n" letras (onde "n" é o número fornecido junto a opção "-n" na linha de comando).

Portanto, para cada grupo de letras processado, serão obtidos quatro valores, cada um destes correspondendo a uma das bases nitrogenadas.

Por exemplo. Supondo que na linha de comando tenha sido fornecida a opção "-n 10". Isso significa que os grupos deverão tamanho de 10 bases (letras). Ainda, suponha que um destes grupos de bases obtido do arquivo de entrada seja o seguinte: "TATACCGCAA". Como resultado da contagem deverá ser obtido os seguintes valores:

- Número de bases "A": 4
- Número de bases "T": 2
- Número de bases "C": 3
- Número de bases "G": 1

Observar que o número de bases (letras) que estão no arquivo de entrada deve ser, no mínimo, igual ao parâmetro fornecido na opção "-n". Se isso não ocorrer, então o arquivo de entrada não tem informações suficientes para ser processado (ver situações de erro na sessão 3)

Estas quatro contagens devem ser escritas no arquivo de saída, conforme está descrito na sessão 6.

Mas, o arquivo de entrada poderá ter mais bases do que o tamanho especificado pela opção "-n". Então, neste caso, o programa deverá contar as bases de vários grupos. Neste caso, é necessário definir onde iniciam cada um dos grupos a serem processados.

Supondo que o arquivo de entrada possua "m" bases, os grupo a serem processados vão iniciar nas posições 0, 1, 2, ..., m-n do arquivo de entrada.

Por exemplo, suponha que foi utilizada a opção "-n 5" (portanto, os grupos terão tamanho 5) e que foi encontrado no arquivo de entrada a seguinte sequência de bases: "TATACCGCA" (portanto, m=9). Nesse caso, o programa deverá processar grupos que iniciam na posição 0 (zero), até o grupo que inicia na posição "m-n=4" (quadro). Estes grupos serão os seguintes:

- Início na posição 0: "TATAC"
- Início na posição 1: "ATACC"
- Início na posição 2: "TACCG"
- Início na posição 3: "ACCGC"
- Início na posição 4: "CCGCA"

Portanto, no caso de exemplo, o programa deverá obter quatro valores para cada um dos 5 (cinco) grupos. Esses valores deverão ser colocados no arquivo de saída, conforme está descrito na sessão 6.

5. Resumo na tela

Se não for encontrado nenhum erro, seu programa deve colocar na tela as informações das opções que estão sendo usadas no processamento e as informações relativas aos dados encontrados no arquivo de entrada.

As informações das opções são as seguintes:

- Nome do arquivo de entrada;
- Nome do arquivo de saída;
- Tamanho dos grupos de bases a serem calculados (valor da opção "-n");
- Informações a serem colocadas no arquivo de saída ("A", "T", "C", "G" e/ou "A+T;C+G").

As informações do arquivo de entrada são as seguintes:

- Número de bases no arquivo de entrada;
- Número de grupos a serem processados;
- Número de linhas do arquivo de entrada que contém bases.

6. Formato do arquivo de saída

O resultado com os contadores das bases nos grupos de bases que estão no arquivo de entrada deverão ser escritos no arquivo de saída da seguinte forma:

A primeira linha do arquivo de saída deve conter um cabeçalho, informando quais são os contadores que estão disponíveis no arquivo. Devem ser usadas as letras “A”, “T”, “C” e “G”, nesta ordem, quando forem colocados no arquivo de saída os contadores de bases “A”, “T”, “C” e “G”, respectivamente.

No caso da opção “-+”, deve ser colocado no arquivo de saída, logo após as informações anteriores, o somatório das contagem de “A” e “T” assim como de “C” e “G”. Neste caso, o cabeçalho deve ser “A+T” e “C+G”. Estas letras devem estar separadas por um “;” (ponto-e-vírgula).

Por exemplo, se a opção da linha de comando foi “-ac+”, então devem ser colocados no arquivo de saída as informações da base “A”, da base “C” e o somatório das bases. Portanto, o cabeçalho será:

```
A;C;A+T;C+G
```

A partir da segunda linha, o programa deverá escrever o conjunto de contadores correspondentes aos grupos, na mesma ordem usada no cabeçalho. Logo, se houver 5 grupos de bases, serão escritas no arquivo de saída 5 linhas, iniciando pela segunda linha e terminando na sexta linha.

7. Entregáveis: o que deve ser entregue?

Deverá ser entregue, via Moodle da disciplina, **APENAS** o arquivo fonte com a solução do problema apresentado, escrito *na linguagem simbólica de montagem* dos processadores 80X86 da Intel (arquivo .ASM). Além disso, esse programa fonte deverá conter comentários descritivos da implementação.

Para a correção, o programa será montado usando o montador **MASM 6.11** no ambiente **DosBox 0.74** e executado com diferentes arquivos e frases de entrada. A nota final do trabalho será proporcional às funcionalidades que forem atendidas pelo programa.

O trabalho deverá ser entregue até a data prevista, conforme programado no MOODLE. **Não será aceita a entrega de trabalhos após a data estabelecida**.

8. Observações

Recomenda-se a troca de ideias entre os alunos. Entretanto, a identificação de cópias de trabalhos acarretará na aplicação do Código Disciplinar Discente e a tomada das medidas cabíveis para essa situação (**tanto o trabalho original quanto os copiados receberão nota zero**).

O professor da disciplina reserva-se o direito, caso necessário, de solicitar uma demonstração do programa, onde o aluno será arguido sobre o trabalho como um todo. Nesse caso, a nota final do trabalho levará em consideração o resultado da demonstração.

9. Como obter a linha de comando?

O string escrito na chamada "linha de comando" pode ser lido por um programa escrito em assembler. Esse string está no PSP – Program Segment Prefix, que se encontra em um segmento específico da memória. Nesse segmento, o string pode ser encontrado a partir do offset 81H. O final do string é identificado pelo byte CR (0DH).

Ainda, no offset 80H pode-se encontrar o tamanho do string digitado na linha de comando, em bytes.

O segmento onde se encontra o PSP está presente nos registradores DS e ES, logo no início da execução do programa.

Entretanto, quando se usa o modo simplificado do MASM (com as diretivas "ponto"), o DS será carregado com o segmento de dados do programa. Assim, a informação do PSP só estará presente no registrador ES.

Portanto, se for desejado usar o ES sem perder as informação do PSP, é necessário salvar a informação do ES, antes de realizar qualquer alteração no ES.

Exemplo: No exemplo abaixo está sendo considerado que o programa será montado pelo MASM no modelo simplificado. Esse trecho de programa deve ser colocado no início do programa pois ele utiliza os valores fornecidos pelo Sistema Operacional nos registradores de segmento DS e ES. A instrução "rep movsb" é responsável por copiar a linha de comando do PSP para uma variável de nome "VAR" no segmento de dados do programa.

```
        push    ds          ; salva as informações de segmentos
        push    es

        mov     ax,ds       ; troca DS <-> ES, para poder usa o MOVSB
        mov     bx,es
        mov     ds,bx
        mov     es,ax

        mov     si,80h      ; obtém o tamanho do string e coloca em CX
        mov     ch,0
        mov     cl,[si]

        mov     si,81h      ; inicializa o ponteiro de origem

        lea     di,VAR      ; inicializa o ponteiro de destino

        rep     movsb

        pop     es          ; retorna as informações dos registradores de segmentos
        pop     ds
```