

Trabalho de Programação 3

Processador Intel (80x86)

1. Descrição Geral

Neste trabalho você deverá desenvolver um programa capaz de ler um arquivo em formato texto (arquivo de entrada), processar os dados lidos e, ao final, escrever um arquivo com os resultados obtidos (arquivo de saída).

No arquivo de entrada estará disponível uma lista de números que representam valores financeiros. O programa deverá ler esses valores e armazená-los na memória. Para isso, sugere-se o formato de ponto fixo (ver Cap. 7 do Livro do prof. Weber – 3ª Edição).

Depois de lidos os números, o programa deverá calcular o dígito verificador de cada número, a soma dos valores e a média aritmética dos mesmos.

Finalmente, os valores lidos e calculados deverão ser escritos no arquivo de saída.

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

2. Arquivo de Entrada

O nome do arquivo de entrada deverá ser fornecido via teclado pelo usuário para o programa. O nome desse arquivo deverá ser um nome válido do DOS. Esse nome de arquivo pode ser formado por um NOME (até 8 caracteres) seguido de um TIPO (até 3 caracteres), separados por um "." (ponto). Exemplo:

arquivo.txt

O nome do arquivo fornecido também poderá ser formado sem o TIPO. Exemplo:

arquivo

No arquivo de entrada estará disponível uma lista de números que representam valores financeiros. Essa lista será formada por no mínimo de 1 (um) valor e o máximo de 100 (cem) valores numéricos.

Cada linha do arquivo de entrada conterá um único valor. Cada valor é representado por sua parte inteira e parte fracionária, separadas por vírgula (",") ou ponto ("."). Uma linha de texto termina quando for encontrado um caractere CR - *Carriage Return* (0DH) ou LF - *Line Feed* (0AH), ou um deles seguido do outro.

A parte inteira poderá ser qualquer valor entre 0 (zero) e 499 (quatrocentos e noventa e nove). A parte fracionária poderá ser qualquer valor entre 0 (zero) e 99 (noventa e nove).

Observar que o final dos dados é identificado através da identificação do final do arquivo.

Caso alguma linha contenha informações inválidas, a linha toda deverá ser ignorada.

Somente são válidas as linhas que contenham valores numéricos, separados por vírgula ou ponto, com os valores inteiro e fracionário dentro dos limites estabelecidos. Pode haver caracteres SPACE (20H) ou TAB (09H) à esquerda ou a direita do número.

Exemplo de arquivo de entrada:

```
Números com vírgula
10,94
120,32
410,12
    21,35
Números com ponto
100.10
99.99
Números inválidos
510,23
100.201
```

3. Processamento

Depois de lidos os números, o programa deverá efetuar os seguintes cálculos sobre os valores lidos:

- Dígito verificador, de cada número lido
- Soma dos valores lidos
- Média aritmética dos valores lidos

O dígito verificador é formado a partir do cálculo da de paridade par da parte inteira e da parte fracionária do número lido (ver item 8.9 do Livro do prof. Weber – 3ª Edição). Portanto, o programa deve calcular a paridade par da parte inteira e a paridade par da parte fracionária.

Exemplo, para o caso do número 410,12:

Parte Inteira		
Decimal	410	
Binário	1 1001 1010	
Paridade	1	
Parte Fracionária		
Decimal	12	
Binário	1100	
Paridade	0	

O próximo cálculo é a soma dos valores lidos do arquivo de entrada. Observar que um arquivo de entrada pode ter até 100 valores numéricos, cada um deles com valor máximo de 499,99. Assim, a soma poderá ser de, no máximo, 49999,00.

Finalmente, para o cálculo da média aritmética dos valores lidos deve-se dividir a soma desses valores pelo número de valores lidos. O resultado da divisão deve ser representado pela sua parte inteira e parte fracionária. O resultado da divisão deve ser arredondado para o valor mais próximo, de maneira que a parte fracionária tenha dois dígitos decimais.

Exemplos:

	Exemplo 1	Exemplo 2
Resultado da divisão	214,235	127,234
Resultado arredondado	214,24	127,23

4. Arquivo de Saída

O nome do arquivo de saída deverá ser formado pela concatenação do nome do arquivo de entrada e o TIPO “RES”. Exemplo: Se o nome do arquivo de entrada for “arquivo.txt”, o nome do arquivo de saída deverá ser:

arquivo.res

Outro exemplo: Se o nome do arquivo de entrada for “arquivo”, o nome do arquivo de saída deverá ser:

arquivo.res

Após calculados todos os valores, os dados de entrada assim como os valores calculados devem ser escritos em um arquivo de saída.

Cada linha do arquivo de saída deve receber um contador, o valor numérico e os dígitos verificadores, separados por um hífen (“-”). O contador inicia com “001” e termina, no máximo, com “100”. O valor numérico deve ser escrito com suas partes componentes separadas por vírgula (“,”). Os dois dígitos verificadores (0 ou 1) devem ser escritos lado a lado. Primeiro (mais à esquerda) o correspondente à parte inteira, e depois o correspondente à parte fracionária. Todos os valores devem estar dispostos de maneira que suas vírgulas estejam verticalmente alinhadas.

Ao final das linhas com os valores numéricos lidos, devem ser escritas duas linhas: uma delas com a soma e outra com a média aritmética calculada. À esquerda de cada valor deve ser acrescentadas as palavras “Soma: ” e “Media: ”, respectivamente.

Exemplo de arquivo de saída:

```
001 - 10,94 - 01
002 - 120,32 - 01
003 - 410,12 - 10
004 - 21,35 - 11
005 - 100,10 - 10
006 - 99,99 - 00
Soma: 762,82
Media: 127,14
```

5. Entregáveis: o que deve ser entregue?

Deverá ser entregue, via Moodle da disciplina, 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, os programas serão montados usando o montador **MASM 6.11** no ambiente **DosBox 0.74** e executados com diferentes arquivos de dados 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.**

6. 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.