

# EA872 Laboratório de Programação de Software Básico

## Atividade 2 (a e b)



**Vinícius Esperança Mantovani**

**RA 247395**

Entrega (limite): 09/08/2023, 16:00

### Exercício a)

- 1) Tem-se de utilizar o “.” para rodar os executáveis em questão por conta de seus caminhos não estarem na variável PATH. Isso porque, ao tentar encontrar o arquivo em diretórios cujos caminhos estão em PATH, o shell não encontra o executável que queremos rodar, pois seu caminho não está salvo nessa variável. Por isso, usamos o “.” para expressar o caminho do diretório atual e a barra para expressar que o arquivo está neste diretório e, o nome do executável. Por isso, também, os executáveis como “gcc” não precisam de “.” para executar, uma vez que ele se encontra em um diretório cujo caminho está em PATH.
- 2) Isso pode ser feito de acordo com o que se segue nas imagens abaixo:

```
ea202@le27-7:~$ PATH=/home/ea202/arqlab1/atividades:$PATH
ea202@le27-7:~$ echo $PATH
/home/ea202/arqlab1/atividades:/usr/local/sbin:/usr/local/
```

- 3) A imagem abaixo traz uma pequena amostra do funcionamento do p\_a, que transforma as letras minúsculas em letras maiúsculas. Isso é feito de modo que, ao receber a entrada do terminal, o p\_a.l tem como primeira regra implementada uma que trata da presença de uma sequência de letras minúsculas ([a-z]). Nesta regra, temos o seguinte: [a-z] printf("%c", yytext[0]-('a'-'A'));

Neste comando, caso haja uma sequência de caracteres alfabéticos minúsculos, no printf, essa sequência é pega do vetor yytext (caracter por caracter) e os converte em caracteres alfabéticos maiúsculos.

```
ea202@le27-7:~/arquivos_apoio_lex$ p_a
abacate456
ABACATE456
```

### Exercício b)

- 1) O programa converte a data do sistema para o sistema de horários AMPM. Para tanto, o programa tem regras que identificam os casos em que se tem “AM” seguido por um espaço e um número de dois caracteres ou “PM”, um espaço e um número de dois caracteres. No caso de AM, estão registrados os números de 00 à 11 e, no caso de PM, estão registrados os números de 12 à 23. Então, na sessão de rotinas, esses casos são tratados de forma que caso o número seja 0 ou 12 eles são convertidos para 12, AM ou PM, da maneira adequada e, caso o número seja outro entre os possíveis, printa no formato certo alterando o valor para valores sempre entre 1 e 12.

```
ea202@le27-7:~/arquivos_apoio_lex$ date | p_b
qua 09 ago 2023 PM 04:02:18 -03
ea202@le27-7:~/arquivos_apoio_lex$ date
qua 09 ago 2023 16:02:24 -03
```

Para alterar os valores, temos no código o uso de yytext para escanear o valor inteiro da variável “h” que contém o valor da hora coletado. Em seguida, já tendo identificado se é o caso “PM” ou “AM”, ocorre a impressão usando as condicionais descritas anteriormente.

Apenas é alterado o campo das horas, os minutos e segundos são preservados.

- 2) O comando pipe faz com que a saída do comando à esquerda seja usado como entrada do comando à direita.
- 3) Outra forma de passar a saída de date para p\_b seria escrever sua saída em um arquivo txt e passar como entrada em p\_b da forma como se apresenta na imagem

```
13 ea202@le27-7:~/arquivos_apoio_lex$ date >> date.txt
14 ea202@le27-7:~/arquivos_apoio_lex$ p_b < date.txt
qua 09 ago 2023 PM 04:24:07 -03
```