

-----oOo-----

## Phần 1: Sử dụng awk để làm việc với file text

AWK là một ngôn ngữ lập trình hướng dữ liệu, thường được dùng cho việc xử lý dữ liệu text dựa trên việc tìm kiếm mẫu dữ liệu. Dữ liệu đầu vào được chia thành các bản ghi (dòng), mỗi bản ghi được chia thành các trường (cột). AWK thường được dùng để lọc và chuẩn hóa dữ liệu đầu ra từ dữ liệu đầu vào ban đầu.

Cú pháp:

```
awk 'BEGIN {câu_lệnh_awk} {thân_chương_trình} END {câu_lệnh_awk}'
```

Chúng ta thấy câu lệnh AWK chia thành các khối

### Khối BEGIN

Cú pháp

```
BEGIN { awk_commands }
```

Khối này chỉ được chạy duy nhất một lần lúc bắt đầu, trước khi awk thực thi khối body cho tất cả các dòng trong file input

- Khối này không bắt buộc
- Từ khóa BEGIN bắt buộc phải viết hoa
- Có thể cho nhiều lệnh vào trong khối BEGIN
- Khối BEGIN này có thể hữu dụng cho việc in các report headers, khởi tạo các biến

### Khối thân chương trình

Cú pháp

```
/pattern/ {action}
```

Khối lệnh trong body được thực thi mỗi lần duyệt một dòng trong input file. Ví dụ nếu trong input file có 10 record thì body sẽ được thực thi 10 lần. Không có từ khóa nào đánh dấu cho khối body này.

## **Khối END**

Cú pháp

*END { awk\_commands }*

Khối END chỉ được thực thi một lần ngay sau khi khối body xử lý xong toàn bộ file input. Một số đặc tính của khối END

- Không bắt buộc
- Từ khóa END phải được viết hoa
- Có thể có nhiều lệnh trong khối END
- Khối này hữu ích cho việc in report footer và làm các thao tác dọn dẹp

## **Một số biến định nghĩa sẵn trong AWK**

\$1: Cột đầu tiên của file

\$2: Cột thứ 2 của file

\$n: Cột thứ n của file

NR: thứ tự hiện tại của bản ghi so với khởi điểm của đầu vào

FILENAME: tên của file đầu vào hiện tại

Ví dụ : File test.txt có nội dung

1	Bill_Gates	k60a2
2	Barack_Obama	k60a3
3	Lionel_Messik	k60a3
4	Kim_Jong_Un	k60a2

VD1: Câu lệnh in ra cột đầu tiên của tất cả các dòng

```
awk '{print $1}' test.txt
```

Kết quả:

1  
2  
3  
4

VD2: Câu lệnh để in ra cột thứ 2 của file test.txt như sau:

```
awk 'BEGIN {print "Ho va ten"} {print $2} END {print "Het"}' test.txt
```

Kết quả:

Ho va ten

Bill\_Gates

Barack\_Obama

Lionel\_Messi

Kim\_Jong\_Un

Het

VD3: Câu lệnh in ra tên file của file test.txt như sau:

```
awk 'BEGIN {print "Noi dung"} {print $2} END {print FILENAME; print "Het"}'  
test.txt
```

Kết quả:

Noi dung

Bill\_Gates

Barack\_Obama

Lionel\_Messi

Kim\_Jong\_Un

test.txt

Het

### **Cấu trúc điều khiển trong AWK**

Awk cũng có các cấu trúc điều khiển như if...else, for, while... giống các ngôn ngữ lập trình khác

VD4: Câu lệnh để in ra tên những học sinh thuộc lớp k60a2 như sau:

```
awk 'BEGIN {print "Ho va ten"} { if ($3 == "k60a2") print $2} END {print "Het"}'  
test.txt
```

Kết quả:

Ho va ten  
*Bill\_Gates*  
*Kim\_Jong\_Un*  
*Het*

VD5: Câu lệnh để in ra mỗi dòng 4 lần

```
awk 'BEGIN {print "Ket qua"} {for (i=1;i<3;i++) print "In dong ", $1, "lan" ,i} END {print "Het"}}' test.txt
```

Hoặc

```
awk 'BEGIN {i=1; print "Ket qua"} {while (i<3){print "In dong ", $1, "lan" ,i; i++}} END {print "Het"}}' test.txt
```

Kết quả:

*Ket qua*  
*In dong 1 lần thứ 1*  
*In dong 1 lần thứ 2*  
*In dong 2 lần thứ 1*  
*In dong 2 lần thứ 2*  
*In dong 3 lần thứ 1*  
*In dong 3 lần thứ 2*  
*In dong 4 lần thứ 1*  
*In dong 4 lần thứ 2*  
*Het*

### **Các hàm cơ bản trong Awk**

Hàm lấy độ dài của chuỗi (length)

VD6: Lấy ra độ dài của tên các học sinh

```
awk 'BEGIN {print "Ket qua"} {print length($2)} END {print "Het"}}' test.txt
```

Kết quả:

*10*  
*12*  
*12*  
*11*

Hàm viết hoa hoặc viết thường cả chuỗi (toupper/tolower)

VD7:

```
awk 'BEGIN {print "Ket qua"} {print toupper($2)} END {print "Het"}' test.txt
```

Kết quả:

*Ket qua*

*BILL\_GATE*

*BARACK\_OBAMA*

*LIONEL\_MESSI*

*KIM\_JONG\_UN*

*Het*

```
awk 'BEGIN {print "Ket qua"} {print toupper($2)} END {print "Het"}' test.txt
```

Kết quả:

*Ket qua*

*bill\_gates*

*barack\_obama*

*lionel\_messi*

*kim\_jong\_un*

*Het*

Hàm cắt chuỗi **substr**

Cú pháp: *substr(chuỗi\_đầu\_vào, cắt\_từ\_vị\_trí, cắt\_đến\_vị\_trí)*

VD: Cắt từ đầu đến ký tự thứ 4 của tên học sinh

```
awk 'BEGIN {print "Ket qua"} {print substr($2,0,4)} END {print "Het"}' test.txt
```

Kết quả:

*Ket qua*

*Bill*

*Bara*

*Lion*

*Kim\_*

*Het*

Hàm lấy ra số thứ tự của ký tự muốn tìm đầu tiên trong chuỗi (INDEX)

Cú pháp: *index(chuỗi\_đầu\_vào, ký\_tự\_muốn\_tìm\_vị\_trí)*

VD: Tìm vị trí của ký tự l trong chuỗi họ tên sinh viên:

```
awk 'BEGIN {print "Ket qua"} {print index($2,"l")} END {print "Het"}' test.txt
```

Kết quả:

*Ket qua*

3

0

1

0

*Het*

## **Phần 2: Bài tập thực hành**

**Bài 1:** Cho file diemso.txt như sau:

1 *Nguyen\_Van\_A* 5 6 7

2 *Pham\_Thi\_B* 6 5 4

3 *Nguyen\_Van\_C* 9 6 8

3 cột cuối là điểm của 3 môn Toán, Lý, Hóa. Hãy dùng awk để hiển thị điểm trung bình của các học sinh như sau:

Diem trung binh

*Nguyen\_Van\_A* 6

*Pham\_Thi\_B* 5

*Nguyen\_Van\_C* 7.6666

Ket thuc

**Bài 2:**Hiển thị họ của tất cả các học sinh trong lớp.

**Bài 3:** Cho file sinhvien.txt có nội dung như sau:

*Sau day la thong tin cac sinh vien trong lop K60A2*

1 *Nguyen\_Van\_A* *Thaibinh*

2 *Pham\_Thi\_B* *Namdinh*

3 *Nguyen\_Van\_C* *Thanhhoa*

Hãy hiển thị họ và tên các sinh viên trong lớp K60A2

## **Phần 3: Liên lạc**

STT	Họ và tên	Email	ĐT
1	Hà Mỹ Linh	Halinh.hus@gmail.com	
3	Trần Thị Hương	<a href="mailto:tranthihuong.hus@gmail.com">tranthihuong.hus@gmail.com</a>	