**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**TÀI LIỆU:**

**HƯỚNG DẪN THỰC HÀNH**

**HỆ ĐIỀU HÀNH**

**Nhóm biên soạn:**

- ThS. Phan Đình Duy

- ThS. Phạm Văn Phước

- ThS. Nguyễn Việt Quốc

- KS. Nguyễn Hữu Nhân

- KS. Lê Văn La

- KS. Trần Văn Quang

***Tháng 3 năm 2015***

**NỘI DUNG CÁC BÀI THỰC HÀNH**

**Phần 1: Lập trình trên Linux**

Bài 1: Hướng dẫn cài đặt Ubuntu và các lệnh cơ bản của shell

Bài 2: Cơ bản lập trình shell

**Phần 2: Thực hành hệ điều hành**

Bài 3: Quản lý tiến trình

Bài 4: Định thời CPU

Bài 5: Đồng bộ hóa tiến trình, tiểu trình

**Bài 6: Quản lý bộ nhớ**

**Phần 3: Bài tập lớn**

CÁC PHẦN MỀM THIẾT BỊ SỬ DỤNG TRONG MÔN THỰC HÀNH

- Phần mềm VMware

- Hệ điều hành Ubuntu

**Bài 6: QUẢN LÝ BỘ NHỚ**

1. **MỤC TIÊU:**

- Sinh viên nắm rõ được việc quản lý bộ nhớ như thế nào, các giải thuật thay thế trang.

- Đánh giá ưu nhược điểm giữa các giải thuật thay thế trang.

- Viết được chương trình mô phỏng các giải thuật thay thế trang bằng c trên linux.

1. **NỘI DUNG THỰC HÀNH**
2. Cho biết khi nào thì xảy ra lỗi trang? Vẽ sơ đồ hoạt động của Hệ điều hành khi xảy ra lỗi trang? Trình bày 2 kỹ thuật cài đặt bộ nhớ ảo **Demand Paging** và **Segmentation Paging**?
3. Tại sao phải thực hiện chiến lược thay thế trang?
4. Vẽ sơ đồ trình bày một số giải thuật thay thế trang sau: FIFO, OPT(optimal), LRU với chuỗi tham chiếu :0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1. Giả sử có 3 khung trang và các khung trang ban đầu là trống. Xác định số **Page Fault** từ đó đưa ra đánh giá các giải thuật.
5. Sử dụng ngôn ngữ lập trình C viết chương trình mô phỏng các giải thuật thay thế trang đã nêu ở mục 3 với các yêu cầu như sau:

* Lúc mới chạy chương trình hiển thị menu như sau:

Trong đó: Chuỗi tham chiếu mặc định là chuỗi cho mục 3.

* Sau khi lựa chọn chuỗi tham chiếu chương trình yêu cầu người dùng nhập số khung trang



* Sau khi nhập số khung trang chương trình chuyển sang menu lựa chọn các giải thuật thay thế trang như sau:



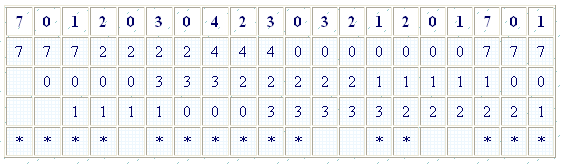
* Sau khi lựa chọn giải thuật để thực hiện thay thế trang chương trình sẽ hiển thị kết quả như sau:

VD: Có chuỗi tham chiếu : 1,2,3,4,1,2,5,1,2,3,4,5. Có 3 khung trang và chọn giải thuật FIFO chương trình hiển thị Output như sau:

1. **SINH VIÊN CHUẨN BỊ:**
2. Sinh viên chuẩn bị ở nhà các mục 1 đến 3 trong phần nội dung thực hành.
3. Sinh viên chạy đoạn code sau mô phỏng giải thuật thay thế trang FIFO xem chương trình chạy như thế nào? Vẽ lưu đồ giải thuật?

| #include<stdio.h>  int main()  {  int i, j, n, a[50], frames[5], frame, k, available, count = 0;  printf(" \nNhập số phần tử chuỗi tham chiếu: \n");  scanf("%d",&n);  printf(" \nNhập vào chuỗi tham chiếu: \n");  for(i=1;i<=n;i++)  scanf("%d",&a[i]);  printf("\nNhập vào số khung trang :\n");  scanf("%d",&frame);  for(i=0;i<frame;i++)  frames[i]= -1; // Giả sử ban đầu các frame trống  j=0;  printf("\t|Chuỗi|\t|Khung trang");  for(k=0;k<frame-1;k++)1  printf("\t");  printf("|\n");  for(i=1;i<=n;i++)  {  printf("\t| %d |\t",a[i]);  available=0; // trang không có sẵn  for(k=0;k<frame;k++)  if(frames[k]==a[i]) // kiểm tra trang có sẵn  available=1; // trang có sẵn  if (available==0) // thay thế trang nếu không có sẵn  {  frames[j]=a[i];  j=(j+1)%frame;  count++;  printf("|");  for(k=0;k<frame;k++)  printf("%d\t",frames[k]);  printf("| F"); // Dấu hiệu nhận biết xảy ra lỗi trang  }  else  {  printf("|");  for(k=0;k<frame;k++)  printf("%d\t",frames[k]);  printf("|");  }  printf("\n");  }  printf("Số trang lỗi là: %d\n",count);  return 0;  } |
| --- |

Chạy đoạn code trên với chuỗi tham số như sau:



1. Sinh viên tìm hiểu các giải thuật còn lại (OTP, LRU) và thực hiện như yêu cầu của **mục 4.**

**--HẾT--**