Câu 1: Danh sách đặc là một danh sách mà các phần tử trong danh sách có cùng kiểu dữ liệu, và được cấp phát liên tục trong bộ nhớ.

Ví dụ: Hàm nhập của danh sách đặc:

void Nhap(int a[],int n)

{

for(int i=0;i<n;i++)

{

cout<<”Nhap a[”<<i<<”]= ”;

cin>>a[i];

}

}

Câu 2:

Danh sách liên kết là danh sách mà các phần tử được cấp phát rời rạc trong bộ nhớ

Danh sách liên kết trong chương trình học được chia là 2 loại là danh sách liên kết đơn và danh sách liên kết kép.

Danh sách liên kết đơn là một danh sách mà trong đo các phần tử được cấp phất rời rạc nhau và cố định trong bộ nhớ. Mỗi phần tử trong danh sách gồm có 2 phần:

+Phần 1: vùng thông tin chứa giá trị cần quản lí (infor)

+Phần 2: vùng liên kết, chứa địa chỉ bộ nhớ của phần tử kế tiếp.(link)

Ví dụ: Cấu trúc của một danh sách liên kết đơn:

struct Node

{

int infor;

Node\* link;

}

Node\* first;

Danh sách liên kết kép là một danh sách mà trong đó mỗi phần tử gồm 3 thành phần:

+Phần 1: vùng liên kết (previous) trỏ đến phần tử đứng liền trước nó.

+Phần 2: vùng liên kết( next ) trỏ đến phần tử đứng liền sau nó.

+Phần 3: vùng thông tin chứa giá trị cần quản lí (infor)

Ví dụ:

struct Node

{

int infor;

Node\* next,\*previous;

}

Node\* first;

Câu 3:

Stack và Queue là danh sách hạn chế vì 2 danh sách đó là một danh sách các phần tử được quản lí theo thứ tự.

Ví dụ:

* Đối với STACK: Phần tử được thêm vào ngăn xếp sau sẽ được lấy ra (xoá) khỏi ngăn xếp trước (Last In First Out).
* Đối với QUEUE: Phần từ được thêm vào trước sẽ được lấy ra(xoá) khỏi trước ( First In First Out ).

Câu 4:

* LIFO: Phần tử được thêm vào ngăn xếp sau sẽ được lấy ra (xoá) khỏi ngăn xếp trước

Ví dụ**:**

**\* Thêm một phần tử vào stack**

int Push(int a[], int &sp, int x)

{

if (sp< MAX -1)

{

a[++sp] = x; return 1;

}

return 0;

}

**\*Lấy ra một phần tử:**

int Pop(int &x)

{

if (sp!= NULL)

{

Node \*p = sp;

x = p -> info;

sp= p ->link;

delete p;

return 1;

}

return 0;

}

* FIFO: Phần tử được thêm vào trước sẽ được lấy ra trước.

Ví dụ:

**\*Thêm một phần tử:**

void Push(int x)

{

Node \*p;

p = new Node;

p->info = x;

p->link = NULL;

if (rear == NULL)

front =p;

else

rear->link = p;

rear = p;

}

**\*Lấy ra một phần tử:**

int Pop(int &x)

{

if (front!= NULL)

{

Node \*p = front;

x = p -> info;

front = front ->link;

if (front == NULL)

rear = NULL;

delete p;

return 1;

}

return 0;

}

Câu 6: Danh sách liên kết thường được dùng để quản lí một danh sách mà tổng số phần tử trong danh sách có thể khá lớn, và không thể biết trước.

Câu 7: Cấu trúc dữ liệu động được cấp phát lúc chạy chương trình, các phần từ nằm rải rác ở nhiều nơi trong bộ nhớ, kích thước danh sách chỉ bị giới hạn do RAM, thao tác thêm xoá đơn giản.

Ví dụ: Danh sách liên kết.