#define Maxversize 10

typedef struct Arcnode {

int adjvex;

struct Arcnode\* next;

}Arcnode;

typedef struct Vnode {

char data;

Arcnode\* first;

}Vnode, Adjlist[Maxversize];

typedef struct {

Adjlist V;

int vexnum, arcnum;

}Graph;

bool visited[Maxversize];

/\*

0->1->4 a->b->e

1->0->3 b->a->d

2->4 c->e

3->1->4 d->b->e

4->0->2->3 e->a->b->c

\*/

void CreatGraph(Graph& G) {

for (int i = 0; i < Maxversize; i++) {

G.V[i].data = 'a'+i;

G.V[i].first = NULL;

}

Arcnode\* p;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 1;

G.V[0].first = p;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 4;

G.V[0].first->next = p;

p->next = NULL;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 0;

G.V[1].first = p;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 3;

G.V[1].first->next = p;

p->next = NULL;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 3;

G.V[2].first = p;

p->next = NULL;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 1;

G.V[3].first = p;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 4;

G.V[3].first->next = p;

p->next = NULL;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 0;

G.V[4].first = p;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 2;

G.V[4].first->next = p;

p = (Arcnode\*)malloc(sizeof(Arcnode));

p->adjvex = 3;

G.V[4].first->next->next = p;

p->next = NULL;

}

typedef struct {

char Vex[Maxsize];

int Edge[Maxsize][Maxsize];

int vexnum, arcnum;

}Mgraph;