1.

typedef struct node

{

int data;

struct node \*next;

union val

{

char \*a;

}e;

}Node;

int main()

{

Node \*a1 = NULL,\*b=NULL;

Node a = { 1,a1,"Abi" };

Node d = { 2,b,"Lak" };

a1 = &d;

b = &a;

printf("%d %d %d %d", a.e.a, d.data++, ++a1->data, b->data);

return 0;

}

2.

typedef struct node

{

int data;

struct node \*next;

union val

{

char \*a;

}e;

}Node;

int main()

{

Node \*a1 = NULL,\*b=NULL;

Node a = { 1,a1,"Abi" };

Node d = { 2,b,"Lak" };

a1 = &d;

b = &a;

printf("%d %d %d %d", a->e.a, d->data++, ++a1.data, b.data);

return 0;

}

3.

typedef struct node

{

int data;

struct node \*next;

union val

{

char \*a;

}e;

}Node;

int main()

{

Node \*a1 = NULL,\*b=NULL;

Node a = { 1,a1,"Abi" };

Node d = { 2,b,"Lak" };

a1 = &d;

b = &a;

printf("%d %d %d %d", a->next->e.a, d.data++, ++a1->data, b->data);

return 0;

}

4.

typedef struct node

{

int data;

struct node \*next;

union val

{

char \*a;

}e;

}Node;

int main()

{

Node \*a1 = NULL,\*b=NULL;

Node a = { 1,a1,"Abi" };

Node d = { 2,b,"Lak" };

a1 = &d;

b = &a;

a.next = a1;

d.next = b;

printf("%s %d %d %d", a.next->e.a, d.next->next->next->data++, ++a1->next->next

->next->next->data, b->data++);

while (1);

return 0;

}

5.

typedef struct node

{

int data;

struct node \*next;

union val

{

char \*a;

}e;

}Node;

int main()

{

Node \*a1 = NULL, \*b = NULL;

Node a = { 1,a1,"Abi" };

Node d = { 2,b,"Lak" };

a1 = &d;

b = &a;

a.next = a1;

d.next = b;

printf("%c %d %d %d", \*(a.next->e.a), d.next->next->next->data++, ++a1->next

->next->data, b->data);

return 0;

}

6.

typedef struct node

{

int data;

struct node \*next;

struct val

{

char \*a;

}e;

}Node;

int main()

{

Node \*a1 = NULL, \*b = NULL;

Node a = { 1,a1,"Abi" };

Node d = { 2,b,"Lak" };

a1 = &d;

b = &a;

a.next = a1;

d.next = b;

printf("%d %d %d", sizeof(Node), sizeof(a.e), sizeof(\*a1));

return 0;

}

7.

typedef struct node

{

int data;

struct node \*next;

struct val

{

char \*a;

struct node \*pre;

}e;

}Node;

int main()

{

Node \*a1 = NULL, \*b = NULL;

//printf("%d", sizeof(\*a1));

Node a = { 1,a1,"Abi",b };

Node d = { 2,b,"Lak",a1 };

a1 = &d;

b = &a;

a.next = a1;

d.next = b;

a.e.pre = b;

d.e.pre = a1;

printf("%s %d %d %s",a.next->e.pre->e.a,a.e.pre->next->e.pre->data,++d.next->next

->e.pre->data,a1->e.pre->e.a);

return 0;

}

8.

typedef struct node

{

int data;

char \*a;

}Node;

int main()

{

Node a1, b1;

Node \*c;

a1.data = 10;

a1.a = "Hi";

c = &a1;

b1.data = 20;

b1.a = "Hello";

c = &b1;

printf("%d %s\n", c->data, c->a);

printf("%d %s\n", a1.data, a1.a);

printf("%d %s\n", b1.data, b1.a);

return 0;

}

9.

typedef struct node

{

int data=20;

char \*a=”Hi”;

}Node;

int main()

{

Node a1, b1;

Node \*c;

a1.data = 10;

a1.a = "Hi";

c = &a1;

b1.data = 20;

b1.a = "Hello";

c = &b1;

printf("%d %s\n", c->data, c->a);

printf("%d %s\n", a1.data, a1.a);

printf("%d %s\n", b1.data, b1.a);

return 0;

}

10.

typedef struct node

{

int data;

char \*a;

}Node;

int main()

{

Node a1, b1;

Node \*c;

a1.data = 10;

a1.a = "Hi";

c = &a1;

b1.data = 20;

b1.a = "Hello";

c = &b1;

printf("%d %s\n", (\*c).data, (\*c).a);

printf("%d %s", (&b1)->data, (&b1)->a);

return 0;

}