User Defined Type I
Struct

CIS 308

Jorge Valenzuela

KANSAS STATE UNIVERSITY

1

C Programming Language

Key Words in C Language

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
continue	for	signed	void
do	if	static	while
default	goto	sizeof	volatile
const	float	short	unsigned

KANSAS STATE UNIVERSITY

2

Struct

- Declaration

```
struct name {
   type1 name1;
   type2 name2;
   ...
} objList;
```

- Example

```
struct person{
    char name[20];
    int age;
} p1, p2;
```

Variables

- Variables

```
struct person p3;
```

type

KANSAS STATE UNIVERSITY

3

Struct

- *Accessing Struct Fields*

```
struct person bobPerson;
bobPerson.age
bobPerson.name
```


KANSAS STATE UNIVERSITY

4

Struct

- *Accessing `struct` Fields*

```
struct person bobPerson;
bobPerson.age = 20;
bobPerson.name
```


KANSAS STATE UNIVERSITY

5

Struct

- *Accessing Struct Fields*

```
struct person bobPerson;
bobPerson.age = 20;
strcpy(bobPerson.name, "Bob");
```


KANSAS STATE UNIVERSITY

6

Struct

- *Accessing Struct Fields*

```
struct person bobPerson;
bobPerson.age = 20;
strcpy(bobPerson.name, "Bob");


bobPerson.name = "Bob";
```


KANSAS STATE UNIVERSITY

7

Struct

- *Accessing Struct Fields*

```
struct person bobPerson;
bobPerson.age = 20;
strcpy(bobPerson.name, "Bob");


bobPerson.name = "Bob"; // Won't compile
```


KANSAS STATE UNIVERSITY

8

Struct

```
#include <stdio.h>
struct point {
  int x;
  int y;
}; //No variables declared here
double getSlope(int, int, int, int);
double getIntercept(int, int, double);

int main() {
…
  slope = getSlope(p1.x, p1.y, p2.x, p2.y);
  intercept = getIntercept(p1.x, p1.y, slope);
…
return 0;
}
```


KANSAS STATE UNIVERSITY

9

Struct

```
#include <stdio.h>
struct point {
  int x;
  int y;
}; //No variables declared here
double getSlope(int  int  int  int ;
double getIntercept int  int  double ;

int main() {
…
  slope = getSlope(p1.x, p1.y, p2.x, p2.y);
  intercept = getIntercept(p1.x, p1.y, slope);
…
return 0;
}
```


KANSAS STATE UNIVERSITY

10

Struct

```
#include <stdio.h>
struct point {
  int x;
  int y;
}; //No variables declared here
double getSlope(struct point, struct point);
double getIntercept(struct point, double);

int main() {
…
  slope = getSlope(p1, p2);
  intercept = getIntercept(p1, slope);
…
return 0;
}
```


KANSAS STATE UNIVERSITY

11

Struct

```
type name[size]

struct person{
   char name[20];
   int age;
};


struct person group[3];

strcpy(group[0].name, "Tom");
group[0].age = 20;
```


KANSAS STATE UNIVERSITY

12

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;
```


KANSAS STATE UNIVERSITY

13

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;

struct person p1;
strcpy(p1.nmae, Tom);
p1.age = 20;
```


KANSAS STATE UNIVERSITY

14

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;

struct person p1;
strcpy(p1.nmae, Tom);
p1.age = 20;

personPtr = &p1;
```


KANSAS STATE UNIVERSITY

15

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;

personPtr = malloc(sizeof(struct person));
```


KANSAS STATE UNIVERSITY

16

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;

personPtr = malloc(sizeof(struct person))

(*personPtr).age = 18;
```


KANSAS STATE UNIVERSITY

17

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;

(*personPtr).age = 18

*personPtr.age = 18; // BAD!
```


KANSAS STATE UNIVERSITY

18

Struct

```
type* name;

struct person{
   char name[20];
   int age;
};


struct person *personPtr;

(*personPtr).age = 18;
```

→ `personPtr->age = 18;`

Preferred way

KANSAS STATE UNIVERSITY

19

Struct

```
struct node{
   char data[30];
   struct node *next; // pointer to the next node
};


struct node *head;
head = malloc(sizeof(struct node));
```

KANSAS STATE UNIVERSITY

20

Struct

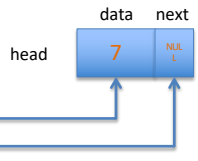
```
struct node{
   int data;
   struct node *next; // pointer to the next node
};

struct node *head;
head = malloc(sizeof(struct node);

head->data = 7;
head->next = NULL;
```

KANSAS STATE UNIVERSITY

21

Struct

```
struct node{
   int data;
   struct node *next; // pointer to the next node
};

struct node *head;
head = malloc(sizeof(struct node);

head->data = 7;
head->next = NULL;
```



KANSAS STATE UNIVERSITY

22

Struct

```
struct node{
   int data;
   struct node *next; // pointer to the next node
};
```

```
struct node *head;
head = malloc(sizeof(structnode);

head->data = 7;
head->next = NULL;

struct node *newNode;
newNode= malloc(sizeof(struct node);
```

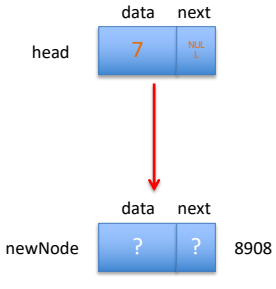
KANSAS STATE UNIVERSITY

23

Struct

```
struct node{
   int data;
   struct node *next; // pointer to the next node
};
```

```
struct node *head;
head = malloc(sizeof(struct node);

head->data = 7;
head->next = NULL;

struct node *newNode;
newNode= malloc(sizeof(struct node);
```



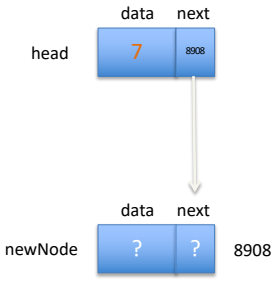
KANSAS STATE UNIVERSITY

24

Struct

```
struct node{
   int data;
   struct node *next; // pointer to the next node
};
```

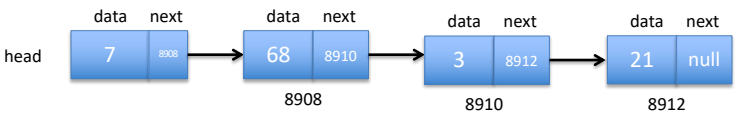
```
struct node *head;
head = malloc(sizeof(struct node);

head->next = newNode;
```



KANSAS STATE UNIVERSITY

25

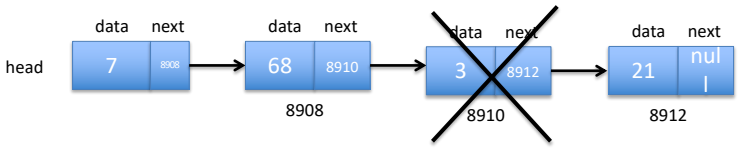
Struct



KANSAS STATE UNIVERSITY

26

Struct

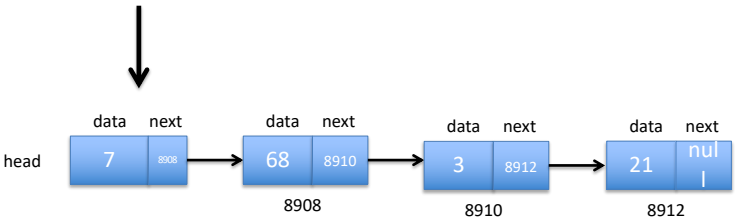


KANSAS STATE UNIVERSITY

27

Struct

remove(3);

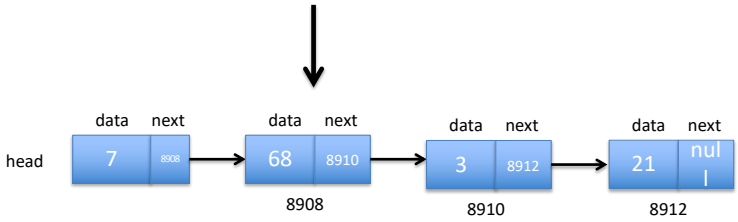


KANSAS STATE UNIVERSITY

28

Struct

remove(3);

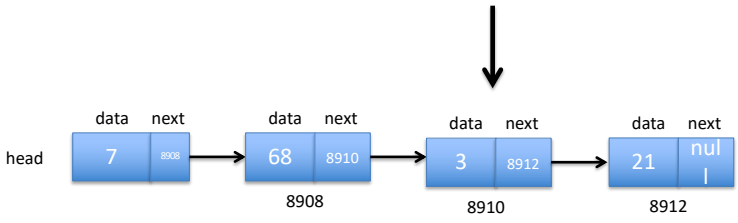


KANSAS STATE UNIVERSITY

29

Struct

remove(3);

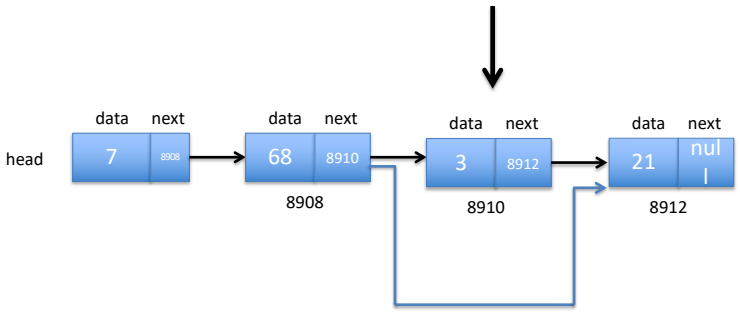


KANSAS STATE UNIVERSITY

30

Struct

remove(3);

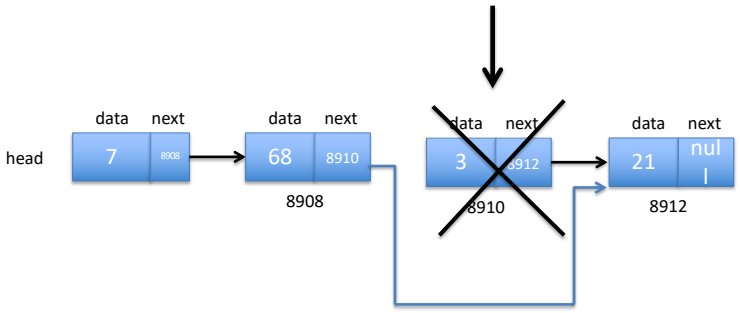


KANSAS STATE UNIVERSITY

31

Struct

remove(3);



KANSAS STATE UNIVERSITY

32

Struct

```
#include <stdlibh>
….
typedef struct node{
   int data;
   Node *next; // pointer to the next node
} Node;


Node*head;
head = malloc(sizeof(Node);

head->data = 7;
head->next = NULL;

Node *newNode;
newNode= malloc(sizeof(Node);
```


KANSAS STATE UNIVERSITY

33

Scanf to read more than one word

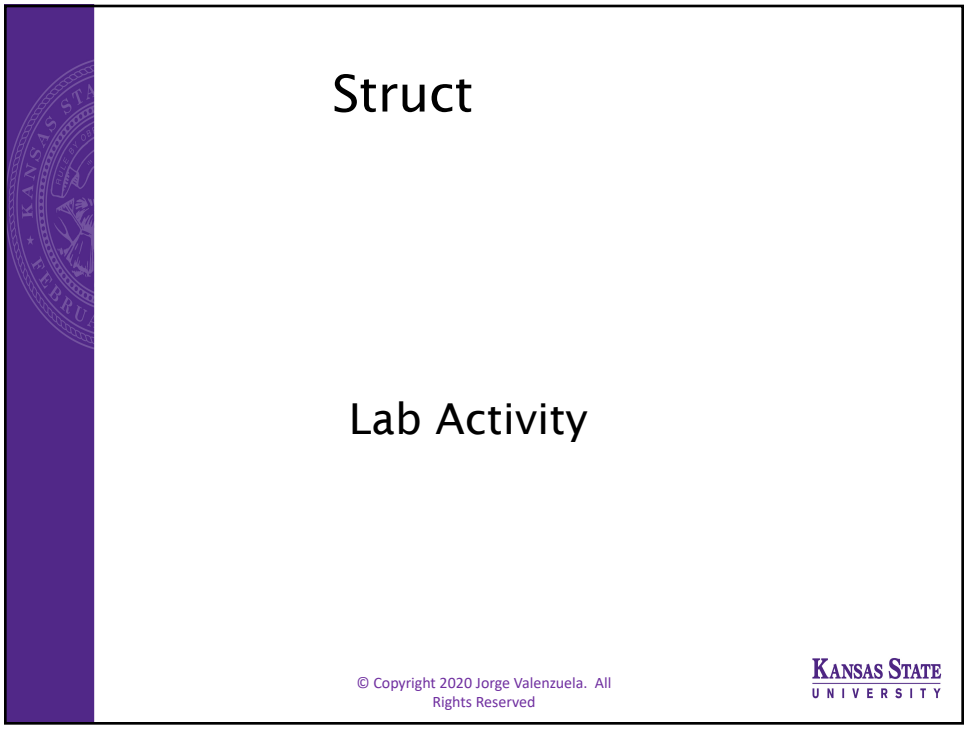
Scanset[]

scanf("%[^\n]s" , name)

Continue to read unless \n encountered

KANSAS STATE UNIVERSITY

34


Struct
Lab Activity
© Copyright 2020 Jorge Valenzuela. All Rights Reserved
KANSAS STATE UNIVERSITY

35