Code:

```c
#include <stdio.h>
#include <stdlib.h>

struct Node{
int data;
struct Node *next;
};

struct Node *head = NULL;

void ins(int position) {
if (position < 1) {
printf("Invalid position\n");
return;
}
struct Node *temp = head;
if (position == 1) {
head = temp->next;
free(temp);
return;
    }

for (int i = 1; i < position - 1 && temp != NULL; i++) {
 temp = temp->next;
}
if (temp == NULL || temp->next == NULL) {
 printf("Position out of range\n");
return;
    }

struct Node *newNode = (struct Node*)malloc(sizeof(struct Node));
newNode->data = 25;
newNode->next = temp->next; temp->next= newNode;
}
int main() {

 struct Node *first, *second;

head = (struct Node*)malloc(sizeof(struct Node));
first = (struct Node*)malloc(sizeof(struct Node));
second = (struct Node*)malloc(sizeof(struct Node));

head->data = 5;
head->next = first;

first->data = 10;
first->next = second;

second->data = 15;
```

```c
return;
}
struct Node *temp = head;
if (position == 1) {
head = temp->next;
free(temp);
return;
    }

for (int i = 1; i < position - 1 && temp != NULL; i++) {
 temp = temp->next;
}
if (temp == NULL || temp->next == NULL) {
 printf("Position out of range\n");
return;
    }

struct Node *newNode = (struct Node*)malloc(sizeof(struct Node));
newNode->data = 25;
newNode->next = temp->next; temp->next= newNode;
}
int main() {

 struct Node *first, *second;

head = (struct Node*)malloc(sizeof(struct Node));
first = (struct Node*)malloc(sizeof(struct Node));
second = (struct Node*)malloc(sizeof(struct Node));

head->data = 5;
head->next = first;

first->data = 10;
first->next = second;

second->data = 15;
second->next = NULL;

ins(2);

struct Node *temp = head;
while(temp != NULL) {
 printf("%d ", temp->data);
 temp = temp->next;
    }

    return 0;
}
```

Output:

```
miruthi@Miruthis-MacBook-Air DSA % gcc insert.c -o insert
miruthi@Miruthis-MacBook-Air DSA % ./insert
5 25 10 15 %
miruthi@Miruthis-MacBook-Air DSA %
```