

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>


#define URL_SIZE 100


// Structure for each browser tab/page
typedef struct Page {
    int pageID;
    char url[URL_SIZE];
    struct Page *prev;
    struct Page *next;
} Page;


Page *head = NULL, *current = NULL;
int idCounter = 1;


// Function to create a new page
Page* createPage(const char *url) {
    Page newPage = (Page)malloc(sizeof(Page));
    newPage->pageID = idCounter++;
    strcpy(newPage->url, url);
    newPage->prev = newPage->next = NULL;
    return newPage;
}


// Visit a new page
void visitNewPage(const char *url) {
```

```
Page *newPage = createPage(url);
if (head == NULL) {
    head = current = newPage;
} else {
    current->next = newPage;
    newPage->prev = current;
    current = newPage;
}
printf("Opened Page %d: %s\n", newPage->pageID, newPage->url);
}
```

// Move to next tab

```
void moveNext() {
    if (current != NULL && current->next != NULL) {
        current = current->next;
        printf("Moved to Next Tab -> Page %d: %s\n", current->pageID, current->url);
    } else {
        printf("No next tab available!\n");
    }
}
```

// Move to previous tab

```
void movePrev() {
    if (current != NULL && current->prev != NULL) {
        current = current->prev;
        printf("Moved to Previous Tab -> Page %d: %s\n", current->pageID, current->url);
    } else {
        printf("No previous tab available!\n");
    }
}
```

```

    }
}

// Show current tab
void showCurrentTab() {
    if (current != NULL) {
        printf("Current Tab -> Page %d: %s\n", current->pageID, current->url);
    } else {
        printf("No page is currently open.\n");
    }
}

// Close a tab by pageID
void closeTab(int id) {
    if (head == NULL) {
        printf("No tabs to close.\n");
        return;
    }

    Page *temp = head;
    while (temp != NULL && temp->pageID != id) {
        temp = temp->next;
    }

    if (temp == NULL) {
        printf("Tab with PageID %d not found!\n", id);
        return;
    }
}

```

```
if (temp == head) {  
    head = temp->next;  
    if (head != NULL) head->prev = NULL;  
} else {  
    temp->prev->next = temp->next;  
}
```

```
if (temp->next != NULL) {  
    temp->next->prev = temp->prev;  
}
```

```
// Adjust current pointer  
if (current == temp) {  
    if (temp->next != NULL)  
        current = temp->next;  
    else  
        current = temp->prev;  
}
```

```
printf("Closed Tab -> Page %d: %s\n", temp->pageID, temp->url);  
free(temp);  
}
```

```
// Show all tabs (history)  
void showHistory() {  
    if (head == NULL) {  
        printf("No tabs available.\n");  
    }
```

```

        return;
    }

    Page *temp = head;

    printf("\n--- Browser Tabs (History) ---\n");

    while (temp != NULL) {

        printf("Page %d: %s", temp->pageID, temp->url);

        if (temp == current) printf(" <-- Current Tab");

        printf("\n");

        temp = temp->next;

    }

    printf("-----\n");
}

```

```

int main() {

    int choice, id;

    char url[URL_SIZE];

    while (1) {

        printf("\n--- Browser Navigation Menu ---\n");

        printf("1. Visit New Page\n");

        printf("2. Move to Next Tab\n");

        printf("3. Move to Previous Tab\n");

        printf("4. Show Current Tab\n");

        printf("5. Close a Tab\n");

        printf("6. Show All Tabs (History)\n");

        printf("7. Exit\n");

        printf("Enter your choice: ");
    }
}

```

```
scanf("%d", &choice);  
getchar(); // consume newline
```

```
switch (choice) {  
    case 1:  
        printf("Enter URL: ");  
        fgets(url, URL_SIZE, stdin);  
        url[strcspn(url, "\n")] = '\0'; // remove newline  
        visitNewPage(url);  
        break;  
    case 2:  
        moveNext();  
        break;  
    case 3:  
        movePrev();  
        break;  
    case 4:  
        showCurrentTab();  
        break;  
    case 5:  
        printf("Enter PageID to close: ");  
        scanf("%d", &id);  
        closeTab(id);  
        break;  
    case 6:  
        showHistory();  
        break;  
    case 7:
```

```
        printf("Exiting Browser...\n");  
        exit(0);  
    default:  
        printf("Invalid choice! Try again.\n");  
    }  
}  
  
return 0;  
}
```