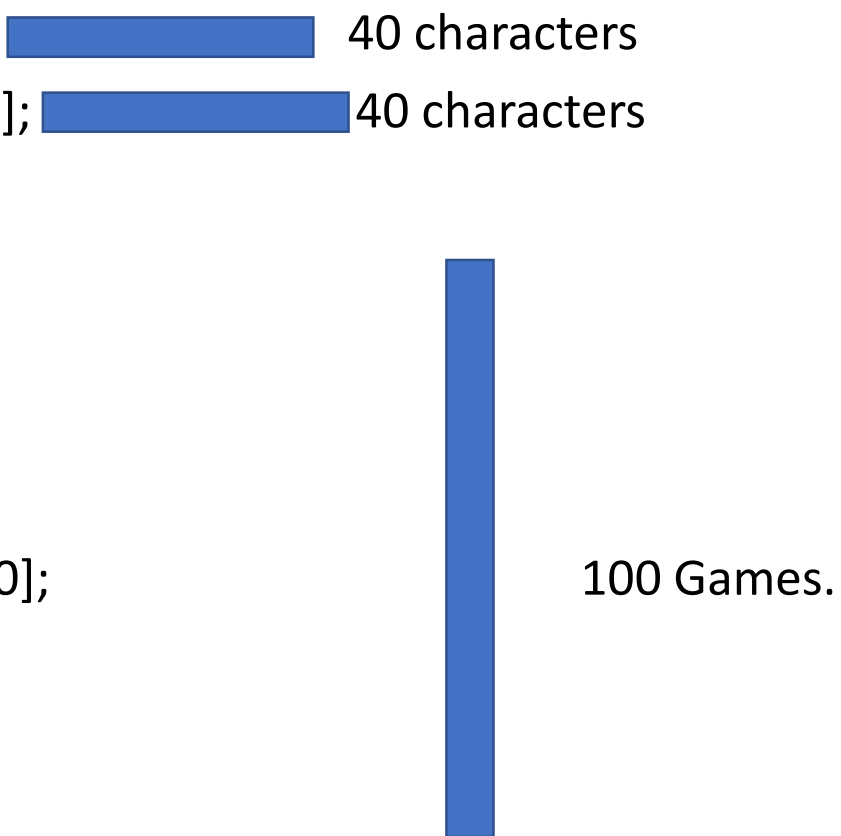# Videogame Catalogue In C

Step by Step

Limitations, Requirements:

- It should store up to 100 games.
- Games have Title, Publisher and an ID.
- We can add games.
- We can view the list of games.
- We can delete a game by it's id.
- We can delete a game by it's location in the list(index).

First, a Structure to Store Games:

```
typedef struct Games{
        char title[40];
        char publisher[40];
        int id;
} Games;

int main()
{
        Games games[100];

}
```

40 characters

40 characters

100 Games.

Add Some Games!!!

Add Some Games!!!

```c
printf("How many games would you like to register? You may enter up to 100 games.\n");
int numberofgames = 0;
scanf("%d", &numberofgames);

    int i = 0;
    while (i < numberofgames)

    {
        printf("Enter the name of the game %d \n", i + 1);
        scanf("%s", games[i].title);

        printf("Enter the publisher of game %d \n", i + 1);
        scanf("%s", games[i].publisher);

        printf("Enter the id of game %d\n", i + 1);
        scanf("%d", games[i].id);


        i = i + 1;
    }
```

View the games!!

View the games!!

```
for(j = 0; j < numberofgames; j++)
{
    printf("Title: %-40s", games[j].title); //-40 is for alignment
    printf("Publisher: %-40s", games[j].publisher);
    printf("Id: %d\n", games[j].id);
}
```

Remove Game by Index

Remove Game by Index

```c
int position = 0;
printf("enter the position of the element you wish to remove\n");
scanf("%d", &position);

int i = 0;
if(position >= numberofgames)
{
    printf("There is no element in entered index.\n");
}
else
{
    for(i=position; i< numberofgames-1 ;i++)
    {
        strcpy(games[i].title, games[i+1].title);
        strcpy(games[i].publisher, games[i+1].publisher);
        games[i].id = games[i+1].id;
    }
    numberofgames = numberofgames-1;
}
```

Remove Game by ID

Remove Game by ID

```c
int enteredid = 0;
printf("enter the id of the element you wish to remove\n");
scanf("%d", &enteredid);
int j = 0;
int i = 0;
int check = 0;

for(j = 0;j< numberofgames; j++)
{
    if(enteredid == games[j].id)
    {
        for(i=j;i< numberofgames;i++)
        {
            strcpy(games[i].title, games[i+1].title);
            strcpy(games[i].publisher, games[i+1].publisher);
            games[i].id = games[i+1].id;
        }
        numberofgames = numberofgames-1;
        check = check + 1;
    }
}

if (check == 0)
{
        printf("A game with such ID is not in the catalogue\n");
}
```
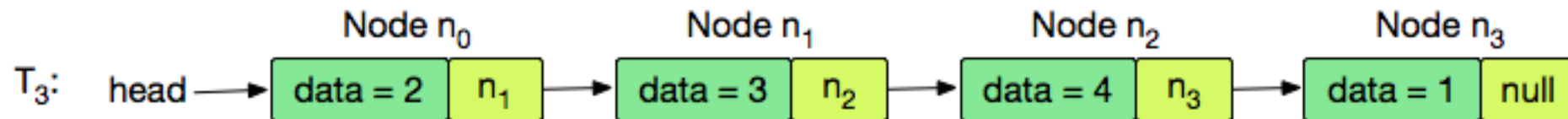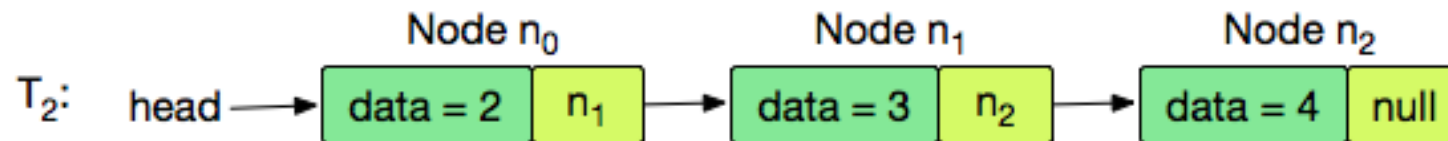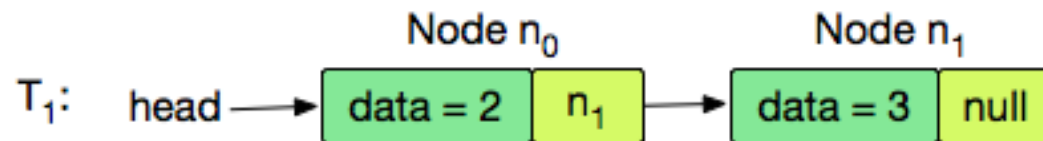
Videogame Catalogue (Linked Lists) in C

Same requirements→

Linked Lists

Initial: head ──► null

Node n_0

T_0:   head ──► | data = 2 | null |

Node n_0 Node n_1

T_1:   head ──► | data = 2 | n_1 | ──► | data = 3 | null |

Node n_0 Node n_1 Node n_2

T_2:   head ──► | data = 2 | n_1 | ──► | data = 3 | n_2 | ──► | data = 4 | null |

Node n_0 Node n_1 Node n_2 Node n_3

T_3:   head ──► | data = 2 | n_1 | ──► | data = 3 | n_2 | ──► | data = 4 | n_3 | ──► | data = 1 | null |

Struct With A Pointer

```c
typedef struct Games {
    int id;
    char title[30];
    char publisher[30];
    struct Games * next;
} games_t;
```

Add Games

```c
//games_t * current = head;
games_t * iter = head;

int max = 0;
printf("How many games would you like to enter? \n");
scanf("%d", &numberofgames);

for(int i = 0; i < numberofgames; i++)
{
    iter->next = malloc(sizeof(games_t));
    printf("Enter the title of the game\n");
    scanf("%s", iter->title);
    printf("Enter the publisher of the game\n");
    scanf("%s", iter->publisher);
    printf("Enter the id of the game\n");
    scanf("%d", &iter->id);
    iter = iter->next;
    //current = current->next;
    iter -> next = NULL;
}
printf("\n\n");
```

View The Games on the List

```c
games_t * current = head;
if(numberofgames == 0)
{
 printf("No game in the list\n\n");
}
for(int k = 0; k < numberofgames; k++)
{
    printf("%-40s", current->title);
    printf("%-40s", current->publisher);
    printf("%d \n", current->id);
    current = current->next;
}
```

Remove Game by Index

```c
games_t * iter = head;

int location = 0;
printf("Enter the index of the element you wish to remove \n");
scanf("%d", &location);

for(int i = 0; i < location; i++)
{
    iter = iter->next;
}
printf("removed the game: \n %-40s", iter ->title);
printf("%-40s", iter ->publisher);
printf("%d \n", iter ->id);
for(int j = location; j < numberofgames; j++)
{
strcpy(iter ->title, iter -> next -> title);
strcpy(iter ->publisher, iter -> next -> publisher);
iter -> id = iter -> next -> id;
}
```

Remove Game by ID

```c
games_t * iter = head;

int deleteid = 0;
printf("Enter the id you wish to remove \n");
scanf("%d", &deleteid);
int loc_count = 0;
for(int i = 0; i < numberofgames; i++)
{
    if(deleteid == iter -> id)
    {
        printf("removed the game: \n %-40s", iter ->title);
        printf("%-40s", iter ->publisher);
        printf("%d \n", iter ->id);

        for(int j = loc_count; j < numberofgames; j++)
        {
            strcpy(iter ->title, iter -> next -> title);
            strcpy(iter ->publisher, iter -> next -> publisher);
            iter -> id = iter -> next -> id;
        }

    }
    iter = iter->next;
    loc_count = loc_count+1;
}
```