#include <stdio.h>

#include <conio.h>

#include <malloc.h>

struct node

{

int info;

struct node \*prev;

struct node \*next;

} \* start;

void createlist(int item)

{ struct node \*ptr,\*new\_node = malloc(sizeof(struct node));

new\_node->info = item;

if (start == NULL)

{ start = new\_node;

new\_node->next = NULL;

new\_node->prev = NULL;

}

else

{ ptr = start;

while (ptr->next != NULL)

ptr = ptr->next;

ptr->next = new\_node;

new\_node->next = NULL;

new\_node->prev = ptr;

}

}

void display()

{

if (start == NULL)

printf("\n Empty List \n");

else

{

struct node \*ptr = start;

while (ptr != NULL)

{

printf("%d ", ptr->info );

ptr = ptr->next;

}

}

}

void main()

{ int data,opt,r,i,loc,c;

char name[10];

struct node \*ptr;

start = NULL;

while (1)

{

printf("\n\n OPTION 1: Creation of New Record.");

printf("\n OPTION 2: Deletion from the beginning of the Record.");

printf("\n OPTION 3: Deletion from the middle of the Record.");

printf("\n OPTION 4: Deletion from the end of the Record.");

printf("\n OPTION 5: EXIT.");

printf("\n\n Enter your option: ");

scanf("%d", &opt);

switch (opt)

{ case 1: printf("\n Enter the no. of Records: ");

scanf("%d", &r);

printf("\n Enter Elements :\n");

for (i = 0; i < r; i++)

{

scanf(" %d", &data);

createlist(data);

}

printf("\n List created succesfully \n");

break;

case 2: if (start == NULL)

{

printf("\n Underflow Error.");

}

else

{

ptr = start;

start = start->next;

start->prev = NULL;

free(ptr);

printf("\n element deleted successfully \n New list is : \n ");

display();

}

break;

case 3: printf("\n Enter the address of element to be deleted from the list :");

scanf("%d", &loc);

if (start == NULL)

printf("\n Underflow Error. ");

else

{ ptr = start;

while (c < loc - 1)

{

ptr = ptr->next;

++c;

}

ptr->next = ptr->next->next;

ptr->next->prev = ptr->next;

}

printf("\n element deleted successfully \n New list is : \n ");

display();

break;

case 4: if (start == NULL)

{

printf("\n Underflow Error. ");

}

else

{ struct node \*temp;

ptr = start;

while (ptr->next != NULL)

ptr = ptr->next;

ptr->prev->next = NULL;

printf("\n element deleted successfully \n New list is : \n ");

display();

}

break;

default: exit(0);

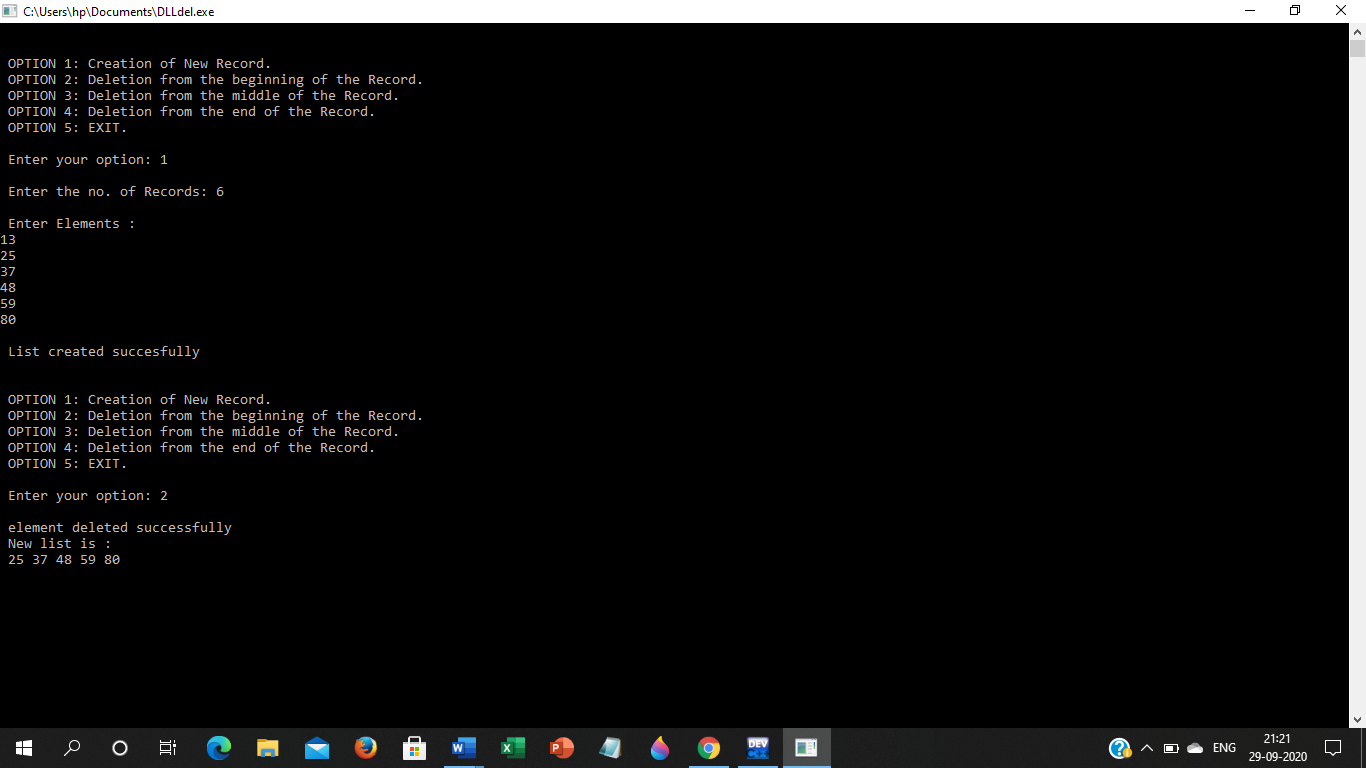
}

getch();

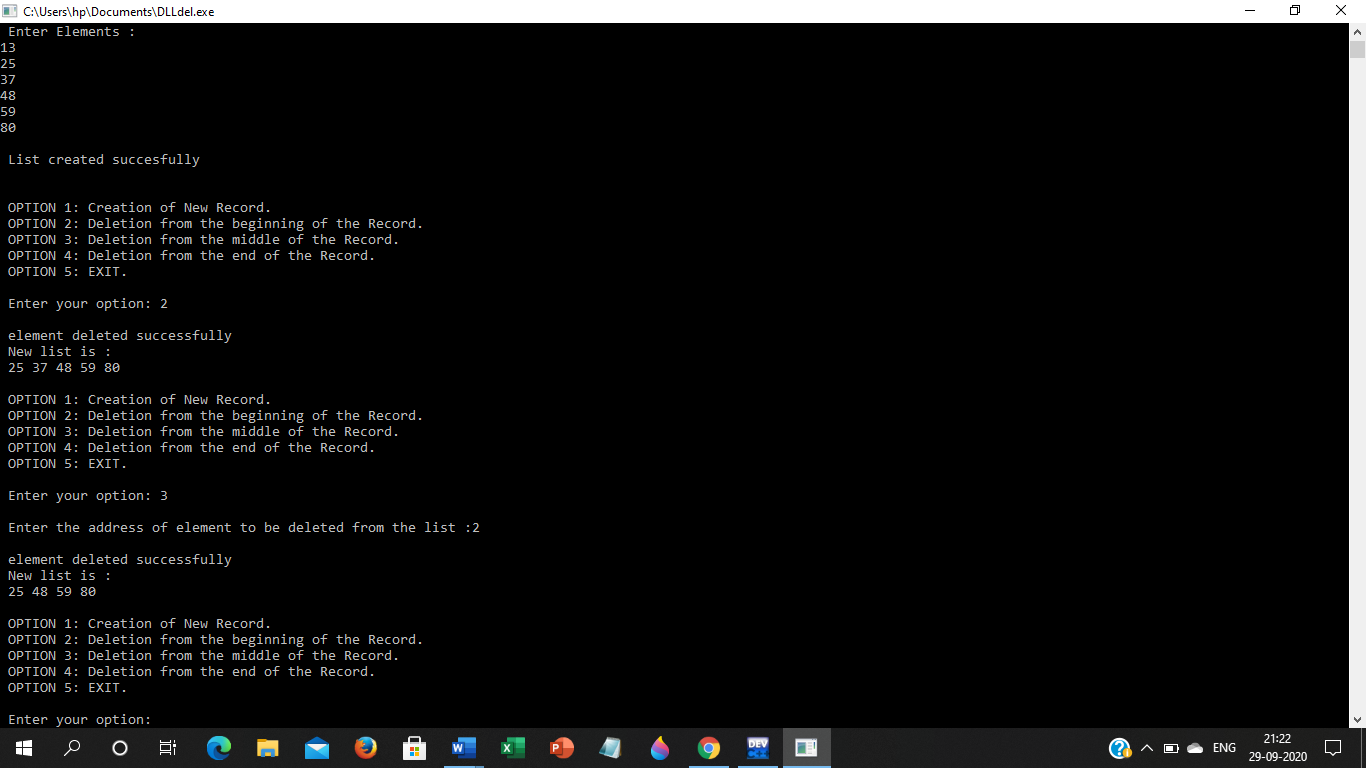
}

}

DELETION FROM BEGINNING:



DELETION FROM SPECIFIC LOCATION:



DELETION FROM END:

