Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

**College of Information Systems and Technology Management**

**2nd Semester A.Y. 2023-2024**

ICC 0104-1 DSA

**GROUP 4**

**Submitted by: Leader**

Garsota, Janine Billones

**Members**

Callangan, Moira

Capiral, Luis

Corales, Charliz Dana

Goyena, Shawn Kieffer

King, Mariano Luiz

Mabutas, Carla

Pajarito, Rose Krishna

Verdida, Maverick

**Submitted to:**

PROF. RICHARD C. REGALA

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

**SOURCE CODE FOR STACK**

| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <ctype.h>  #include <windows.h>  typedef struct STUDREC {  char No[13];  char Name[25];  char Cyr[10];  float GWA;  struct STUDREC \*next;  } STUDREC;  STUDREC \*TOP = NULL;  char Resp;  void clearScreen() {  system("cls");  }  void push(STUDREC \*\*stack, STUDREC \*node) {  node->next = \*stack;  \*stack = node;  }  void deleteTop() {  if (TOP == NULL) {  printf("Stack is empty!\n");  return;  }  STUDREC \*temp = TOP;  TOP = TOP->next;  free(temp);  printf("Top node deleted successfully!\n");  }  void traverseStack() {  STUDREC \*current = TOP;  if (current == NULL) { |
| --- |

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

| printf("The Student Record is Empty!\n");  } else {  clearScreen();  printf("\t\t\t\tStudent Record\n");  printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\n");  printf("| %-12s | %-25s | %-10s | %s |\n", "Number", "Name", "Crs & Yr", "GWA");  printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\n");  while (current != NULL) {  printf("| %-12s | %-25s | %-10s | %.2f |\n", current->No, current->Name, current->Cyr, current->GWA);  current = current->next;  }  printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\n");  }  }  void gotoxy(short x, short y) {  COORD pos = {x, y};  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), pos); }  int main() {  Resp = 'Y';  do {  clearScreen();  printf("\nMain Menu:\n");  printf("1. Create Student Record Stack\n");  printf("2. Add Student Record\n");  printf("3. Delete Top Student Record\n");  printf("4. Display Student Records\n");  printf("5. Exit\n");  printf("\nEnter your choice: "); |
| --- |

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

| int choice;  scanf("%d", &choice);  switch(choice) {  case 1: {  clearScreen();  if (TOP == NULL) {  printf("\nCreating Student Stack...\n");  do {  STUDREC \*NewNode = (STUDREC \*)malloc(sizeof(STUDREC));  if (NewNode == NULL) {  printf("Memory allocation failed\n");  return 1;  }  printf("\nEnter Student Number: ");  scanf("%12s", NewNode->No);  printf("Enter Student Name: ");  getchar();  fgets(NewNode->Name, sizeof(NewNode->Name), stdin);  NewNode->Name[strcspn(NewNode->Name, "\n")] = '\0';  printf("Enter Student Course & Year: ");  fgets(NewNode->Cyr, sizeof(NewNode->Cyr), stdin);  NewNode->Cyr[strcspn(NewNode->Cyr, "\n")] = '\0';  printf("Enter Student GWA: ");  scanf("%f", &NewNode->GWA);  NewNode->next = TOP;  TOP = NewNode;  printf("\nAdd Another Record [Y/N]?: ");  getchar();  scanf("%c", &Resp);  } while (toupper(Resp) == 'Y');  } else {  printf("\nStudent Stack already exists.\n");  }  break;  }  case 2: {  if (TOP == NULL) { |
| --- |

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

| printf("\nPlease create the Student Stack first!\n");  } else {  clearScreen();  do {  STUDREC \*NewNode = (STUDREC \*)malloc(sizeof(STUDREC));  if (NewNode == NULL) {  printf("Memory allocation failed\n");  return 1;  }  printf("\nEnter Student Number: ");  scanf("%12s", NewNode->No);  printf("Enter Student Name: ");  getchar();  fgets(NewNode->Name, sizeof(NewNode->Name), stdin);  NewNode->Name[strcspn(NewNode->Name, "\n")] = '\0';  printf("Enter Student Course & Year: ");  fgets(NewNode->Cyr, sizeof(NewNode->Cyr), stdin);  NewNode->Cyr[strcspn(NewNode->Cyr, "\n")] = '\0';  printf("Enter Student GWA: ");  scanf("%f", &NewNode->GWA);  NewNode->next = TOP;  TOP = NewNode;  printf("\nAdd Another Record [Y/N]?: ");  getchar();  scanf("%c", &Resp);  } while (toupper(Resp) == 'Y');  }  break;  }  case 3: {  if (TOP == NULL) {  printf("\nPlease create the Student Stack first!\n");  } else {  deleteTop();  }  break;  }  case 4: {  if (TOP == NULL) { |
| --- |

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

printf("\nPlease create the Student Stack first!\n"); } else {

traverseStack();

}

break;

}

case 5: {

printf("Exiting program...");

return 0;

}

default: {

printf("Invalid choice! Please try again.");

break;

}

}

printf("\nPress Enter to continue...");

getchar();

getchar();

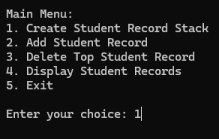
} while (1);

return 0;

}

**SAMPLE OUTPUTS FOR STACKS**

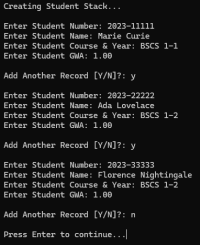
**1. Creation of Stack**

****

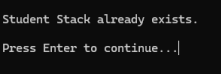
Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

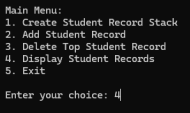
Intramuros, Manila, Philippines



*if you tried to create even if it already exists:*

**

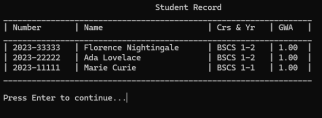
**2. Traversal of Stack**

****

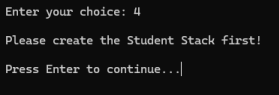
Republic of the Philippines 

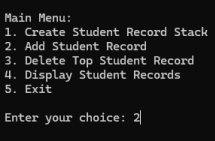
**PAMANTASAN NG LUNGSOD NG MAYNILA**

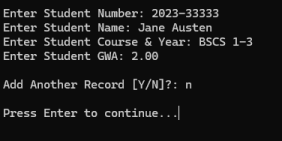
Intramuros, Manila, Philippines



*If no item exists in the stack:*

****3. Adding Node in Stack**

****

****

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

**4. Deleting Node in Stack**

****

****

*If all items are deleted/no element exists:*

**

Republic of the Philippines 

**PAMANTASAN NG LUNGSOD NG MAYNILA**

Intramuros, Manila, Philippines

**5. Exit**

****