Nisarg Wath
642403014

Assignment 3

Convert Binary

Header.h

```
typedef struct Stack{
    char * data;
    int size;
    int top;

}Stack;


void init(Stack *s, int size);
void push(Stack * s, char value);
char pop(Stack *s);
int isEmpty(Stack s);
```

Logic.c

```
#include <stdio.h>
#include <stdlib.h>
#include "header.h"



int isEmpty(Stack s){
    if(s.top == -1) return 1;

    return 0;
}


void init(Stack *s, int size) {
    s->data = (char *)malloc(size * sizeof(char));  // Allocate
memory for 'size' characters
```

```
    s->top = -1;
    s->size = size;
}


void push(Stack *s, char value) {
    if (s->top == s->size - 1) {
        printf("Stack Overflow!!\n");
        return;
    }
    s->data[++(s->top)] = value;
}


char pop(Stack *s) {
    if (s->top == -1) {
        printf("Stack is Empty!\n");
        return '\0';
    }
    return s->data[s->top--];
}


char peek(Stack *s) {
    if (s->top == -1) {
        return '\0';
    }
    return s->data[s->top];
}
```

Main.c

```
#include <stdio.h>
#include <stdlib.h>
#include "header.h"

int main() {
    int n;


    printf("Enter a decimal number: ");
    scanf("%d", &n);

    printf("Binary representation of %d: ", n);
    int str[32];
```

```
    int i = 0;


    if (n == 0) {
        printf("0\n");
        return 0;
    }


    while (n > 0) {
        str[i] = n % 2;
        n = n / 2;
        i++;
    }

    Stack s;
    init(&s, 32);


    for (int j = 0; j < i; j++) {
        push(&s, str[j]);
    }


    printf("Reversed Binary: ");
    while (!isEmpty(s)) {
        printf("%d", pop(&s));
    }
    printf("\n");

    return 0;
}

}
```