**DS Day-01**

**Lab Questions :**

1. **Reversing a 32 bit signed integers:**

#include <stdio.h>

#include <limits.h>

int reverseInteger(int x) {

int reversed = 0;

while (x != 0) {

int pop = x % 10;

x /= 10;

if (reversed > INT\_MAX/10 || (reversed == INT\_MAX / 10 && pop > 7)) return 0;

if (reversed < INT\_MIN/10 || (reversed == INT\_MIN / 10 && pop < -8)) return 0;

reversed = reversed \* 10 + pop;

}

return reversed;

}

int main() {

int num = 12345;

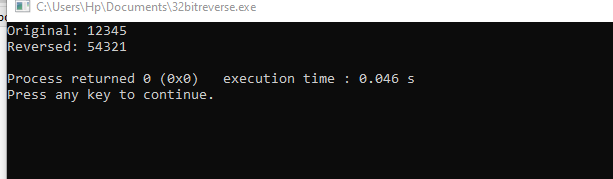
int reversedNum = reverseInteger(num);

printf("Original: %d\n", num);

printf("Reversed: %d\n", reversedNum);

return 0;

}



1. **Check for a valid String:**

#include <stdio.h>

#include <string.h>

int isValidString(char \*str) {

for (int i = 0; i < strlen(str); i++) {

if (!(str[i] >= 'a' && str[i] <= 'z') && !(str[i] >= 'A' && str[i] <= 'Z')) {

return 0;

}

}

return 1;

}

int main() {

char str[] = "ValidString";

if (isValidString(str)) {

printf("The string is valid.\n");

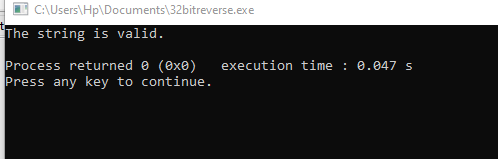
} else {

printf("The string is not valid.\n");

}

return 0;

}

****

1. **Merging two Arrays:**

#include <stdio.h>

int main() {

int arr1[] = {1, 2, 3};

int arr2[] = {4, 5, 6};

int size1 = sizeof(arr1) / sizeof(arr1[0]);

int size2 = sizeof(arr2) / sizeof(arr2[0]);

int size3 = size1 + size2;

int merged[size3];

for (int i = 0; i < size1; i++) {

merged[i] = arr1[i];

}

for (int i = 0; i < size2; i++) {

merged[size1 + i] = arr2[i];

}

printf("Merged Array: ");

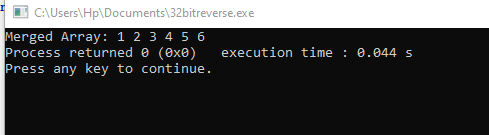
for (int i = 0; i < size3; i++) {

printf("%d ", merged[i]);

}

return 0;

}

****

1. **Given an array finding duplication values:**

#include <stdio.h>

void findDuplicates(int arr[], int size) {

for (int i = 0; i < size; i++) {

for (int j = i + 1; j < size; j++) {

if (arr[i] == arr[j]) {

printf("Duplicate value: %d\n", arr[j]);

}

}

}

}

int main() {

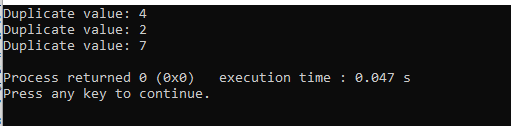
int arr[] = {4, 2, 7, 4, 9, 2, 5, 7};

int size = sizeof(arr) / sizeof(arr[0]);

findDuplicates(arr, size);

return 0;

}

****

1. **Merging of list:**

#include <stdio.h>

int main() {

int list1[] = {1, 3, 5, 7, 9};

int list2[] = {2, 4, 6, 8, 10};

int mergedList[10];

int i, j, k;

for (i = 0; i < 5; i++) {

mergedList[i] = list1[i];

}

for (j = 0; j < 5; j++) {

mergedList[i + j] = list2[j];

}

printf("Merged List: ");

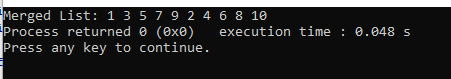
for (k = 0; k < 10; k++) {

printf("%d ", mergedList[k]);

}

return 0;

}

****

1. **Given array of reg nos need to search for particular reg no:**

#include <stdio.h>

int main() {

int regNos[] = {123, 456, 789, 1011, 1213};

int searchRegNo = 1011;

int size = sizeof(regNos) / sizeof(regNos[0]);

int found = 0;

for (int i = 0; i < size; i++) {

if (regNos[i] == searchRegNo) {

found = 1;

break;

}

}

if (found) {

printf("Registration number %d found in the array.", searchRegNo);

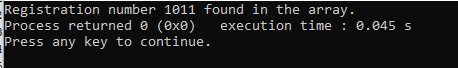
} else {

printf("Registration number %d not found in the array.", searchRegNo);

}

return 0;

}

****

1. **Identify location of element in given array:**

#include <stdio.h>

int findElementIndex(int arr[], int size, int element) {

for (int i = 0; i < size; i++) {

if (arr[i] == element) {

return i;

}

}

return -1; // Element not found

}

int main() {

int arr[] = {10, 20, 30, 40, 50};

int size = sizeof(arr) / sizeof(arr[0]);

int element = 30;

int index = findElementIndex(arr, size, element);

if (index != -1) {

printf("Element %d found at index %d", element, index);

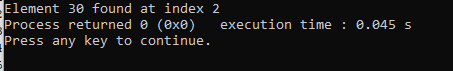
} else {

printf("Element not found in the array");

}

return 0;

}

****

1. **Given array print odd and even values:**

#include <stdio.h>

int main() {

int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

int n = sizeof(arr) / sizeof(arr[0]);

printf("Odd numbers: ");

for (int i = 0; i < n; i++) {

if (arr[i] % 2 != 0) {

printf("%d ", arr[i]);

}

}

printf("\nEven numbers: ");

for (int i = 0; i < n; i++) {

if (arr[i] % 2 == 0) {

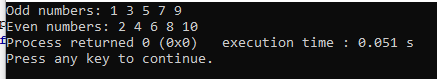
printf("%d ", arr[i]);

}

}

return 0;

}

****

1. **Sum of Fibonacci Series:**

#include <stdio.h>

int main() {

int n, first = 0, second = 1, next, sum = 0;

printf("Enter the number of terms: ");

scanf("%d", &n);

printf("Fibonacci Series: ");

for (int i = 1; i <= n; ++i) {

printf("%d, ", first);

sum += first;

next = first + second;

first = second;

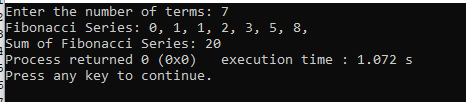
second = next;

}

printf("\nSum of Fibonacci Series: %d", sum);

return 0;

}

****

1. **Finding factorial of a number:**

#include <stdio.h>

int main() {

int number, i;

unsigned long long factorial = 1;

printf("Enter a positive integer: ");

scanf("%d", &number);

if (number < 0)

printf("Error! Factorial of a negative number doesn't exist.");

else {

for (i = 1; i <= number; ++i) {

factorial \*= i;

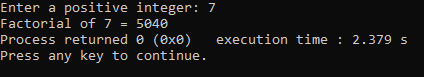
}

printf("Factorial of %d = %llu", number, factorial);

}

return 0;

}

****