**DAY – 11**

Free(): double free detected in tcache 2 --- aborted (core dump)

#include <stdio.h>

#include <stdlib.h>

int main() {

int \*ptr=NULL;

ptr =(int\*)malloc(3\*sizeof(int));

free(ptr);

/\* free(ptr);\*/ ---- if we detect the extra free func we can clear the above error .

printf("\n\n");

return 0;

}

**STRINGS:**

Collection of characters

Ways of declaration & initialization:

Char Name[6] = {‘H’,’e’,’l’,’l’,’o’,\’0’};

Char Name[6] = “Hello “;

Char Greet[] = “Hello World”;

Char Str1[21];

Char Name[][20] = {“ssd”,”wdf”, “cd”}

Column size is necessary not the row

Char Names[10][50]; ---- there are 10 names each of max cap of 50 parts

#include <stdio.h>

int main() {

char Name[5];

int i;

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

scanf("%c",&Name[i]);

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

printf("\n%c=%d",Name[i],Name[i]);

scanf("%s",Name);

printf("%s",Name);

printf("\n");

return 0;

}

#include <stdio.h>

int main() {

char Name[5];

int i;

for(i=0;i<20;i++)

scanf("%c",&Name[i]);

for(i=0;i<20;i++)

printf("\n%c=%d",Name[i],Name[i]);

printf("\n");

scanf("\n%s",Name);

puts(Name);

printf("%s",Name);

printf("\n");

return 0;

}

ERROR: gets the stck smash error as the we are exceeding the string size and in scanf we have done some illegal operation

Destination and source are not equal the destination contain 12

It returns the integer not character ----- strcmp();2

**Memset** --- to initialize the structures ,integers etc;

**Strtok** --- tokenizing the string

Fetch from base address character by charchter

If it stop at any point it again start from their and search for the delimiter

Starting and ending index will change without affecting the original index

**Strlen** --- length of 7string excluding the null

Strerror --- to handle the string handling errors.

Str str ---- to find the first occurrence of string from the main string

Str char ---- to find the first occurrence of character from the main string

Strrchar ----- to find the last occurrence of character from the main string.

**Haystack** ---- finds the first occurrence of substring

#include <stdio.h>

#include<string.h>

e.g.: srtring compare

int main() {

char s1[20],s2[20];

int ret =0;

scanf("%s%s",s1,s2);

printf("\ns1=%s\t2=%s",s1,s2);

ret = strcmp(s1,s2);

printf("\nRet = %d",ret);

return 0;

}

EX: strcpy

#include <stdio.h>

#include<string.h>

int main() {

char s1[20],s2[20];

char d1[40],d2[40];

char \*ptr=NULL;

scanf("%s%s",s1,s2);

strcpy(d1,s1);

printf("\ns1=%s\t2=%s",s1,s2);

/\* ptr = (char\*)malloc(strlen(s1)+1);

ret = strcmp(s1,s2);

printf("\nRet = %d",ret);\*/

printf("\nd1=%s",d1);

strcpy(d1,s2);

printf("\nd1=%s",s2);

return 0;

}

EX1: strtok

#include <stdio.h>

#include<string.h>

int main() {

char s1[400],s2[20];

char \*ptr=NULL;

printf("\nEnter the Line with delimiter\n");

scanf("%[^\n]s",s1);

ptr = strtok(s1," ,");

printf("\nptr = %s",ptr);

printf("\n\n");

return 0;

}

EX:2

#include <stdio.h>

#include<string.h>

int main() {

char s1[400],s2[20];

char \*ptr=NULL;

printf("\nEnter the Line with delimiter\n");

scanf("%[^\n]s",s1);

ptr = strtok(s1," ,");

printf("\nptr = %s",ptr);

ptr = strtok(NULL," ,");

printf("\nptr = %s",ptr);

ptr = strtok(NULL," ,");

printf("\nptr = %s",ptr);

printf("\n\n");

return 0;

}

To reverse a given string

To reverse given intervals of the string

**USER DEFINED DATATYPE:**

1.structures

Struct tagName

{

Member of structure

};

Struct square

{

Int len;

Int breadth;

};

typdef struct Square SQR;

SQR s7,s8,s9;

Typedef struct Square

{

Int len;

Int breadth;

}SQR1;

SQR1 s1;

1. . => static variable => namevar .memberName
2. -> => pointer variable => nameVar->memberName

Ex:

SQR1 \*ptr;

S1.len; --- for normal static variable

Ptr->len; --- for pointer variable

We can’t define the functions in structures.

#include <stdio.h>

#include <string.h>

struct Employee

{

int eId;

char eName[20];

float eSal;

char eGender;

char eAddress[20];

};

typedef struct Employee Emp;

int main(){

Emp e1;

// Emp e[3];

e1.eId = 101;

e1.eSal= 100000.10;

strcpy(e1.eName,"Navya");

strcpy(e1.eAddress,"KAR");

e1.eGender='F';

printf("\nEmployee details are: \n");

printf("\nID: %d",e1.eId);

printf("\nName: %s",e1.eName);

printf("\nSalary: %f",e1.eSal);

printf("\nGender: %c",e1.eGender);

printf("\nAddress: %s",e1.eAddress);

printf("\n\n");

return 0;

2. unions

3.enums