**AIM1:-Five students data should be store in array(ID,Name,Address) and display data value.**

**Algorithm**:-

Step 1- Start

Step 2-initialize struct employee as userdefine datatype & store id,name,address into the particular structure

Step 3-Into main fuction declare struct employee as employees[array size] & i,count as an integers.

Step 4-Input count

Step 5-For i till i<=count go to step 7

Step 6-give input to the employee.id,employee.name,employee.address.

Step 7-i = i +1 go to step 5

Step 8-For i till i<=count go to step 10

Step 9-print employee.id,employee.name,employee.address.

Step 10-i = i + 1 go to step 8

Step 11-Stop

**Program**:-

#include<stdio.h>

struct employee{

int id;

char name[100];

char address[100];

};

int main()

{

struct employee employees[20];

int i,count;

printf("enter count");

scanf("%d",&count);

for(i=1;i<=count;i++)

{

printf("enter id:");

scanf("%d",&employees[i].id);

printf("enter name:");

scanf("%s",&employees[i].name);

printf("enter address:");

scanf("%s",&employees[i].address);

}

for(i=1;i<=count;i++)

{

printf("id:");

printf("%d",employees[i].id);

printf("\n");

printf("name:");

printf("%s",employees[i].name);

printf("\n");

printf("address:");

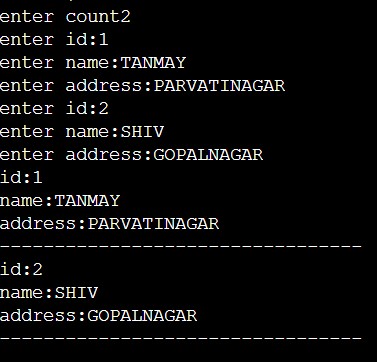
printf("%s",employees[i].address);

printf("---------------------------------");

}

}

**OUTPUT:-**

****

**AIM2 - Sort using bubble sort based based on its employeeid.**

**Algorithm:-**

Step 1-Start

Step 2- initialize struct employee as userdefine datatype & store id,name,address into the particular structure

Step 3- input n

Step 4-for i=0 till i < n go to step6

Step 5-input employee[i].id, employee[i].name, employee[i].add,

Step 6-i=i+1

Step7-for i till i < n go to step 12

Step 8-for j=i+1 till j < n go to step 11

Step 9- if employee[i].id > if employee[j].id go to step 10

Step 10-Swap the two position values

Step 11-j++ go to step 8

Step 12-i++ go to step 7

Step 13-print employee[i].id , employee[i].name,employee[i].add

Step 14-Stop

**Program:-**

#include<stdio.h>

struct emp {

int id;

char name[100];

char add[100];

};

int main()

{

struct emp employee[100];

int i,j,n,temp;

printf("enter the n:");

scanf("%d",&n);

printf("no of iterations are:");

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

{

printf("enter id:");

scanf("%d",&employee[i].id);

printf("enter name:");

scanf("%s",&employee[i].name);

printf("enter add:");

scanf("%s",&employee[i].add);

}

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

{

for(j=i+1;j<n;j++)

{

if(employee[i].id>employee[j].id)

{

temp = employee[i].id;

employee[i].id = employee[j].id;

employee[j].id = temp;

}

}

}

printf("sorted array \n");

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

{

printf("id:");

printf("%d\n",employee[i].id);

printf("name:");

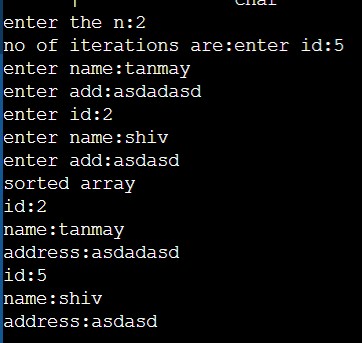
printf("%s\n",employee[i].name);

printf("address:");

printf("%s\n",employee[i].add);

}

}**OUTPUT:-**



AIM3- sort using stack based on its employeeid.

**Algorithm:-**

Step 1-Start

Step 2- declare int a[size], n,i,j,pos,swap as an integers.

Step 3- input n

Step 4-for i=0 till i < n go to step6

Step 5-input array[i]

Step 6-i=i+1

Step7-for i till i < n -1 go to step 12

Step8-make index(i) position default where all comparision results are shown with the help of iterable j variable

Step 9-for j=i+1 till j<n go to step 12

Step 10- if pos!=i go to step 11

Step 11-Swap the two position values

Step 12-i++ go to step 8

Step 13-j++ go to step 7

Step 14-print a[i]

Step 15-Stop

**Program:-**

#include <stdio.h>

int main()

{

int a[100], n, i, j, pos, swap;

printf("Enter number of elements");

scanf("%d", &n);

printf("Enter %d Numbers", n);

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

scanf("%d", &a[i]);

for(i = 0; i < n - 1; i++)

{

pos=i;

for(j = i + 1; j < n; j++)

{

if(a[pos] > a[j])

pos=j;

}

if(pos != i)

{

swap=a[i];

a[i]=a[pos];

a[pos]=swap;

}

}

printf("Sorted employees Id:\n");

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

printf("%d\n", a[i]);

return 0;

}

**Output:-**

