**Question1:**

#include<stdio.h>

int rightrotation(int num,int rot)

{

num=num>>rot;

return num;

}

int main()

{

int num,rotations,res;

printf("enter a num:");

scanf("%d",&num);

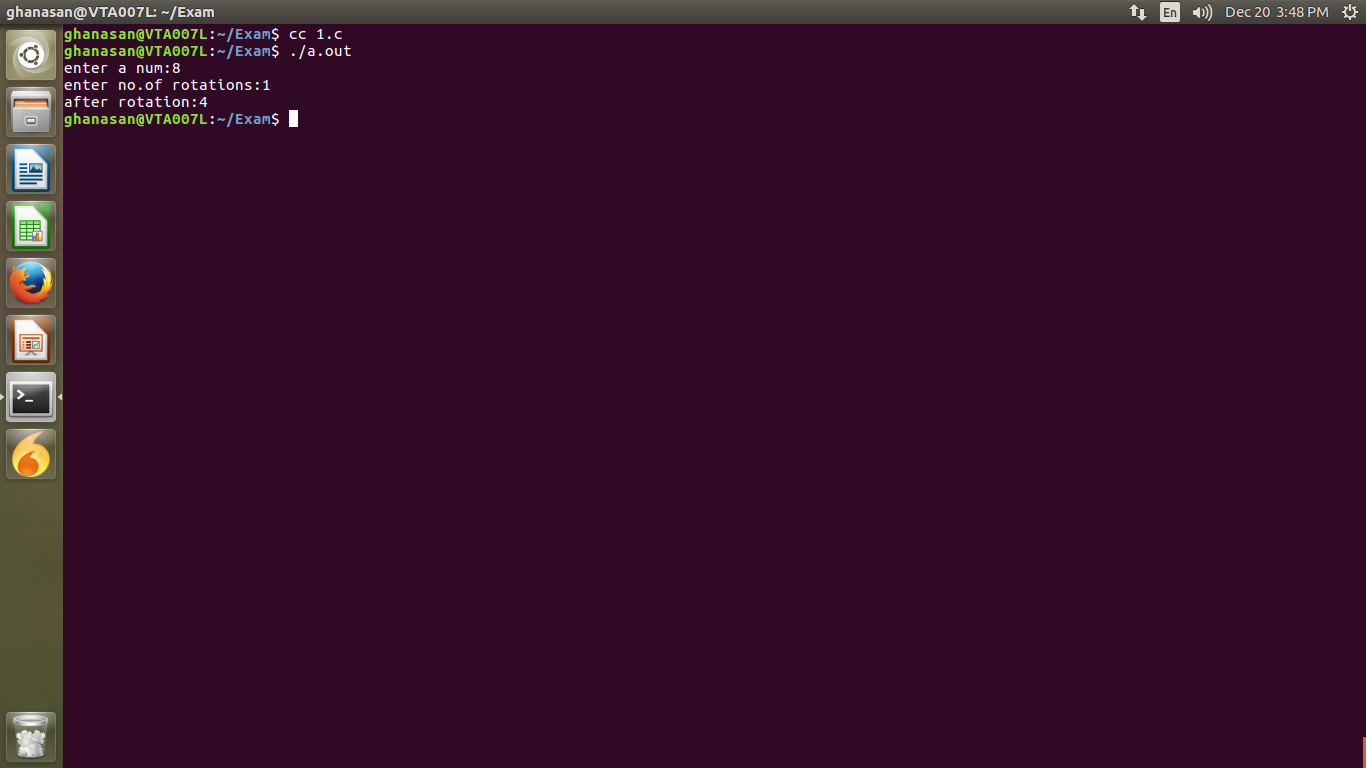
printf("enter no.of rotations:");

scanf("%d",&rotations);

res=rightrotation(num,rotations);

printf("after rotation:%d\n",res);

}



**Question2:**

#include<stdio.h>

int main()

{

int size1,size2;

printf("enter size of arr1:");

scanf("%d",&size1);

printf("enter size of arr2:");

scanf("%d",&size2);

int arr1[size1],arr2[size2];

int ind1,ind2,cnt=0;

printf("enter values of arr1:");

for(ind1=0;ind1<size1;ind1++)

scanf("%d",&arr1[ind1]);

printf("enter values of arr2:");

for(ind1=0;ind1<size2;ind1++)

scanf("%d",&arr2[ind1]);

for(ind1=0;ind1<size2;ind1++)

{

for(ind2=0;ind2<size1;ind2++)

{

if(arr2[ind1]==arr1[ind2])

break;

}

if(ind2==size1)

{

printf("arr2 is not a subset of arr1\n");

return 0;

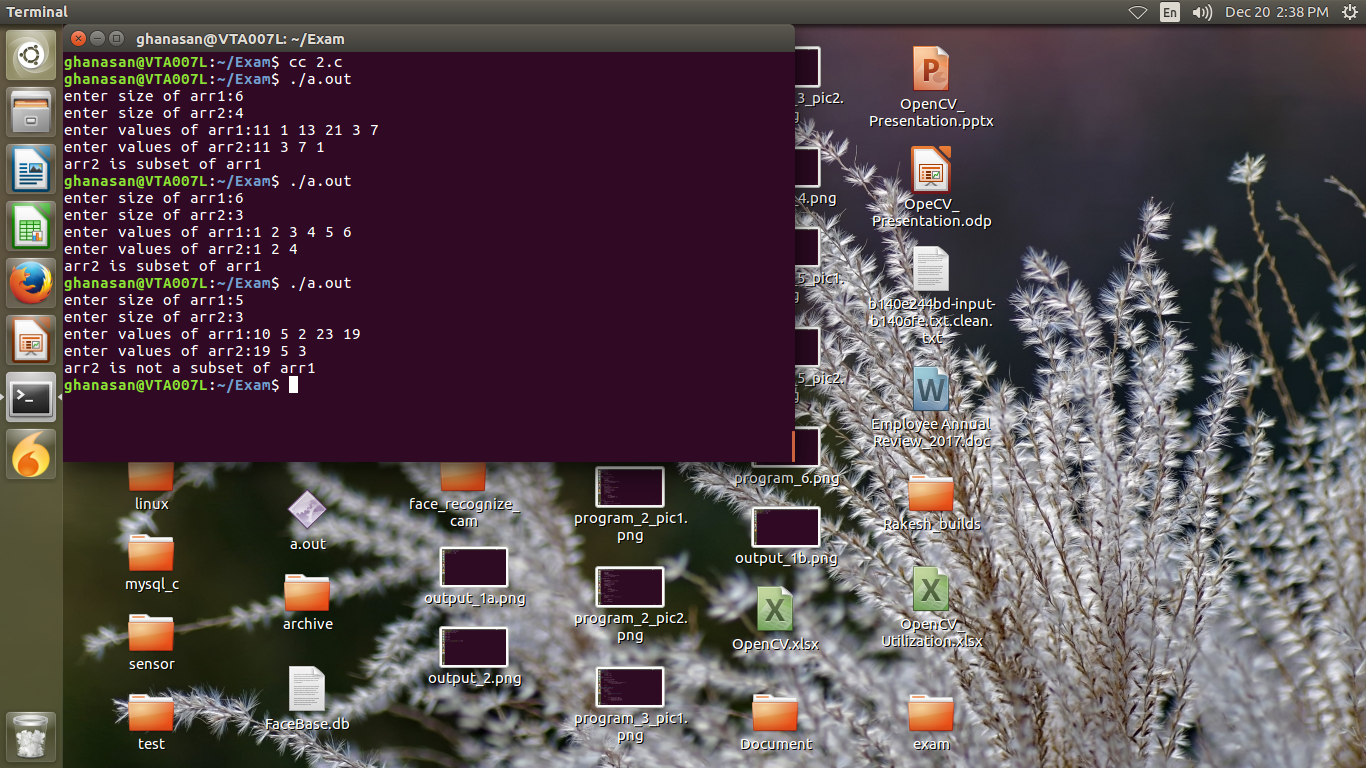
}

}

printf("arr2 is subset of arr1\n");

return 0;

}



**Question3:**

#include<stdio.h>

int main()

{

int arr[10],res[10],ind1,ind2,cnt=0;

puts("enter values into array:");

for(ind1=0;ind1<10;ind1++)

scanf("%d",&arr[ind1]);

for(ind1=0;ind1<10;ind1++)

{

for(ind2=0;ind2<10;ind2++)

{

if((arr[ind1]==arr[ind2])&&(ind1!=ind2))

break;

}

if(ind2==10)

res[cnt++]=arr[ind1];

}

puts("resultant arr is:");

for(ind1=0;ind1<cnt;ind1++)

printf("%d ",res[ind1]);

printf("\n");

}



**Question4:**

#include<stdio.h>

void add(int num1,int num2)

{

int res;

res=num1+num2;

printf("add is:%d\n",res);

}

void sub(int num1,int num2)

{

int res;

res=num1-num2;

printf("sub is:%d\n",res);

}

void mul(int num1,int num2)

{

int res;

res=num1\*num2;

printf("mul is:%d\n",res);

}

void div(int num1,int num2)

{

int res;

res=num1/num2;

printf("div is:%d\n",res);

}

int main()

{

int num1,num2;

printf("enter two nums:");

scanf("%d%d",&num1,&num2);

void (\*ptr)(int,int);

ptr=add;

ptr(num1,num2);

ptr=sub;

ptr(num1,num2);

ptr=mul;

ptr(num1,num2);

ptr=div;

ptr(num1,num2);

}



**Question5:**

#include<stdio.h>

#include<stdlib.h>

struct bst

{

struct bst \*left;

int data;

struct bst \*right;

};

void insert(struct bst \*\*node,int data)

{

struct bst \*nu;

if(\*node==NULL)

{

nu=malloc(sizeof(struct bst));

nu->data=data;

nu->left=NULL;

nu->right=NULL;

\*node=nu;

}

else

{

if(data<(\*node)->data)

insert(&(\*node)->left,data);

else if(data>(\*node)->data)

insert(&(\*node)->right,data);

}

}

void preorder(struct bst \*root)

{

if(root)

{

printf("%d ",root->data);

preorder(root->left);

preorder(root->right);

}

}

void postorder(struct bst \*root)

{

if(root)

{

postorder(root->left);

postorder(root->right);

printf("%d ",root->data);

}

}

void inorder(struct bst \*root)

{

if(root)

{

inorder(root->left);

printf("%d ",root->data);

inorder(root->right);

}

}

int main()

{

struct bst \*root=NULL;

int n,i=0,data;

printf("enter no.of elements:");

scanf("%d",&n);

printf("please enter values is the form of BST tree only\n");

while(i<n)

{

printf("enter data:");

scanf("%d",&data);

insert(&root,data);

i++;

}

printf("preorder:");

preorder(root);

printf("\n");

printf("inorder:");

inorder(root);

printf("\n");

printf("postorder:");

postorder(root);

printf("\n");

}

