Bomb Blast Survivors An N\*N grid is passed as input to the program. There are bombs planted in the grid. Each bomb is detonated and destroys all the blocks in its explosion radius (If the radius is 1, the bomb can destroy all the 8 blocks surrounding it). There are people inside the grid represented by P. The bombs are represented by an integer which is the explosion radius of the bombs. The program must print the number of survivors after the explosion of all bombs. Boundary Condition(s): 1 <= N <= 100 1 <= Explosion radius <= 9 Input Format: The first line contains N The next N lines contain N characters each. Output Format: The first line contains the count of survivors.

**Example Input/Output 1:**

**Input:** 6

\*\*P\*\*\*

\*\*\*\*P\*

\*P\*1\*\*

\*\*\*\*\*P

\*\*\*P\*P

\*\*\*PP2

**Output: 2**

**Explanation:** After explosion the grid becomes (X represents exploded area),

\*P\*\*\*

\*\*XXX\*

\*PX1X\*

\*\*XXXX

\*\*\*XXX

\*\*\*XX2

**Example Input/Output 2:**

**Input:** 6

\*2\*P\*\*

\*\*\*\*\*\*

\*P\*1P\*

\*P\*\*\*\*

\*\*\*\*\*P

\*1\*P\*\*

**Output: 3**

#include<stdio.h>

#include <stdlib.h>

int main()

{

char a[100][100];

int n,index=0;

scanf("%d",&n);

while(scanf("%s",a[index])>0)

{

index++;

}

for(int index=0;index<n;index++)

{

for(int index1=0;index1<n;index1++)

{

if(a[index][index1]>='1' && a[index][index1]<='9')

{

int d=a[index][index1]-48;

for(int index2=index-d;index2<=index+d;index2++)

{

if(index2>=0)

{

for(int index3=index1-d;index3<=index1+d;index3++)

{

if(index3>=0)

{

if(a[index2][index3]>='1' && a[index2][index3]<='9')

{

a[index2][index3]=a[index2][index3];

}

else

{

a[index2][index3]='X';

}

}

}

}

}

}

}

}

int count=0;

for(int index=0;index<n;index++)

{

for(int index1=0;index1<n;index1++)

{

// printf("%c",a[index][index1]);

if(a[index][index1] == 'P')

{

count++;

}

}//printf("\n");

}

printf("%d",count);

}