Qus 3

```
# include <Stdio.h>
# include <String.h>
void cipher(int, int c);
int findMin();
void makeArray(int, int);
char arr[22][22], dara[22][22], message[111],
 retmessage[111], Key[55];

char temp[55], temp2[55];
int K=0,
 int main()
{
  char *message, *dmessage;
int i,j, klen, emlen, flag=0,
inter, c, index, min, rows;
clrscr();
 Printf("Centis the key\n");
 fflush(Stdin);
gets(Key);
 Printf("\n Entis message to be ciphered\n");
 fFlush(Stdin);
gets(message);
Strcpy(temp, Key);
   K=0;
for(i=0; i<klen; ++)
{
  index = findMix();
cipher(index, R);
3
```

```
emessage[K] = '\0';
Printf("\n encryption mussage is \n");
for(i=0; emassage[i] != NULL; i++)
printf("%.c", emessage[i]);
printf("\n\n");
//deciphering
 emlem = strlen(emessage);
//emlem is length of encrypted message.
strpt(temp, Key);
Rows = emlem/Keys
//Row is not of row of the array to made from ciphered
  message rows;
 j=0;
 for(i=0, K=1; emessage[i] != NULL; i++, K++)
{
//printf("\nemessage=%d", emlem);
  temp2[j++] = emessage[i];
if((i+1)%rows)==0))
  {

 temp2[j] = '\0';
inter = findMin();
makeArray(index, Rows);
j=0;
}
}
printf("\n Array Retrive as is\n");
K=0;
for(i=0; i<r; i++)
for(j=0; j<R; j++) {
printf("%c" ,arr[i][j]);
```

```
}
printf("\n");
}
Returnmessag[k]='\0';
printf("\n message retrived is \n");
for(i=0; retmessage[i]!=NULL; i++)
printf("%c", retmessage[i]);
getch();
 Return (0);
{
void cipher(int i, int a) {
int j;
for(j=0;j<t;j++) {
 emessage[k++]=arr[i][i];
}
}
// emessage[k]='\0';
}
void makeArray(int col, int row) {
int i,j;
for(i=0; i<row; i++) {
 arr[i]=(int*)temp2[i];
}
}
 temp[index]=123;
  Return (index);
}
```