

Compute the transitive closure of a given directed graph using Warshall's algorithm

```
#include <stdio.h>
```

```
int n;
```

```
int a[10][10];
```

```
int p[10][10];
```

```
void write_data() {
```

```
    int i, j;
```

```
    printf("The path matrix is shown below\n");
```

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

```
        for (j = 0; j < n; j++) {
```

```
            printf("%d ", p[i][j]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
}
```

```
void read_data() {
```

```
    int i, j;
```

```
    printf("Enter the number of nodes: ");
```

```
    scanf("%d", &n);
```

```
    printf("Enter the adjacency matrix:\n");
```

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

```
        for (j = 0; j < n; j++) {
```

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

```
        }
```

```
    }
```

```
}
```

```

void path_matrix() {
    int i, j, k;
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            p[i][j] = a[i][j];
        }
    }
    for (k = 0; k < n; k++) {
        for (i = 0; i < n; i++) {
            for (j = 0; j < n; j++) {
                if (p[i][k] == 1 && p[k][j] == 1)
                    p[i][j] = 1;
            }
        }
    }
}

```

```

int main() {
    read_data();
    path_matrix();
    write_data();
    return 0;
}

```

Output

Enter the number of nodes: 2

Enter the adjacency matrix:

2

3

3

6

The path matrix is shown below

2 3

3 6

=== Code Execution Successful ===