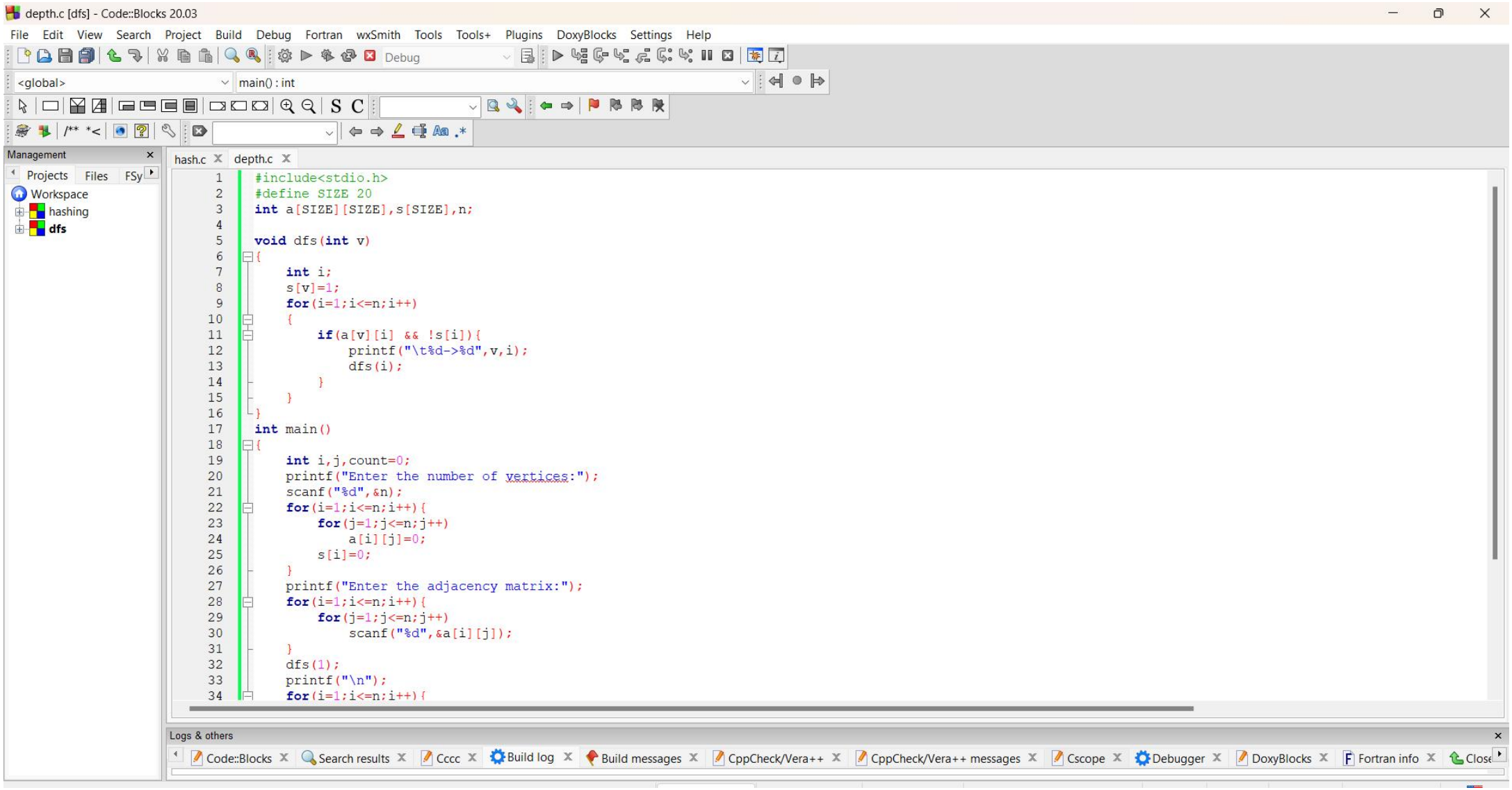
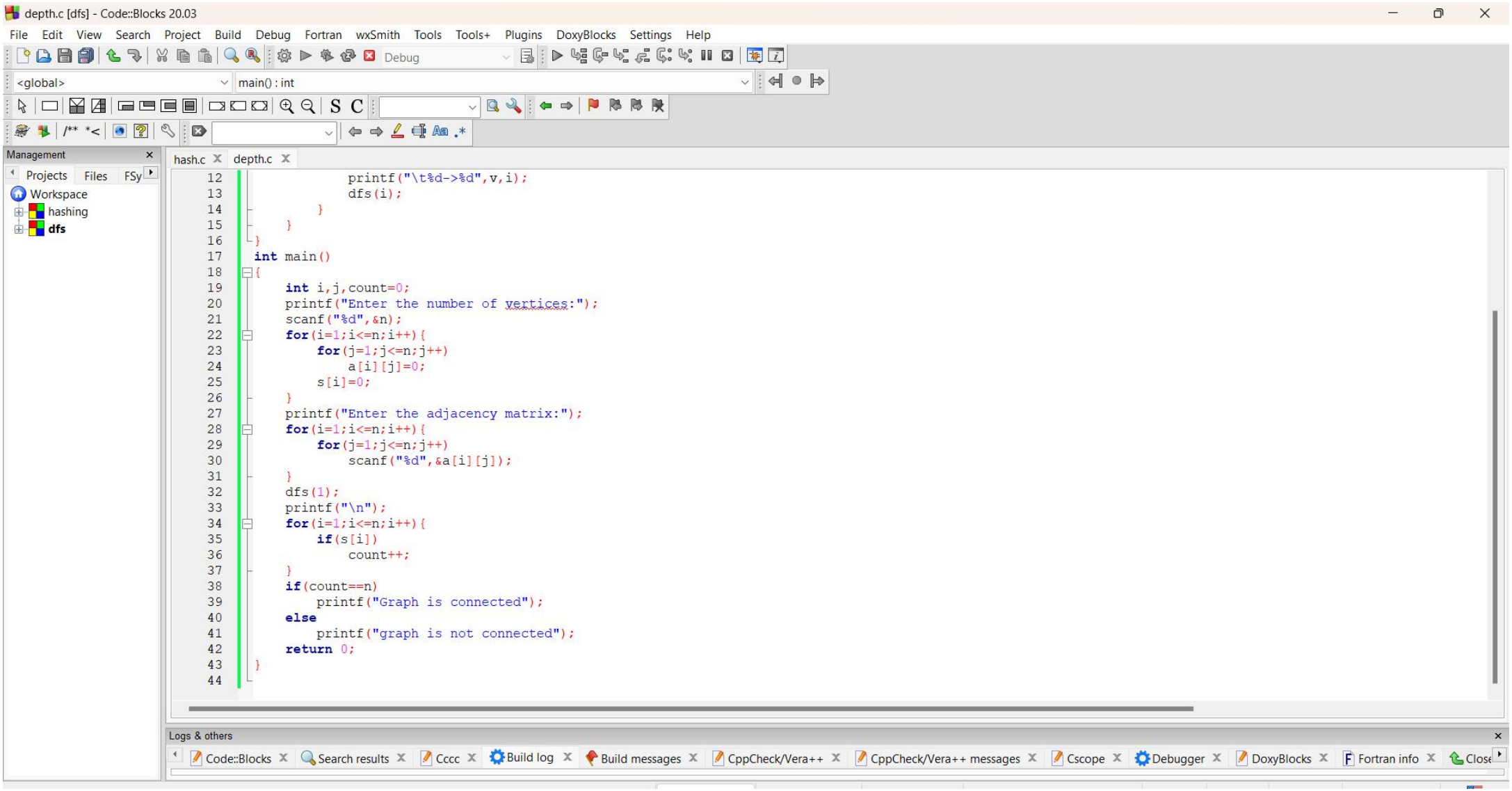
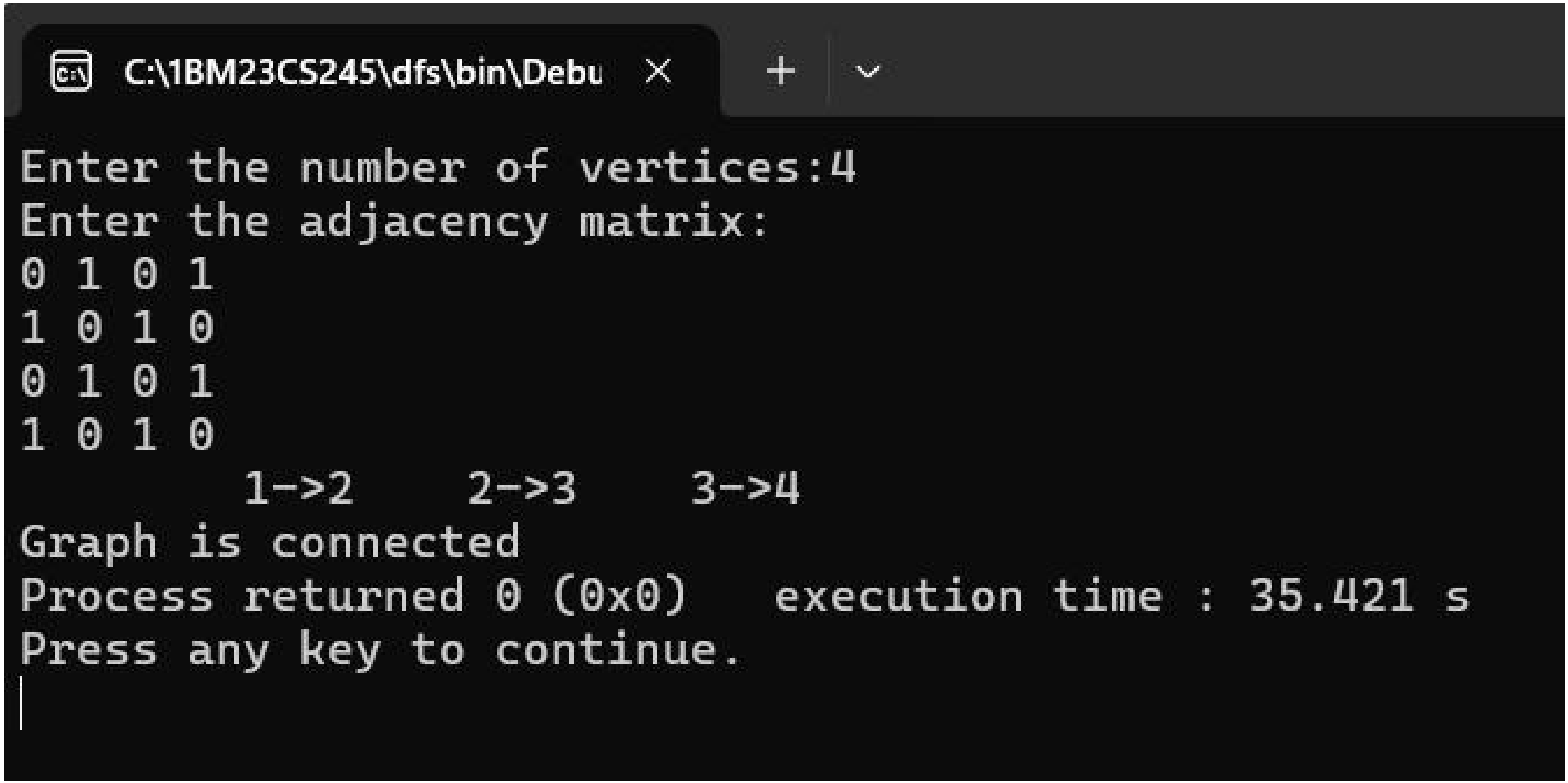

depth.c [dfs] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main() : int
Management
Projects Files FSy
Workspace
hashing
dfs
hash.c
depth.c
1 #include<stdio.h>
2 #define SIZE 20
3 int a[SIZE][SIZE],s[SIZE],n;
4
5 void dfs(int v)
6 {
7 int i;
8 s[v]=1;
9 for(i=1;i<=n;i++)
10 {
11 if(a[v][i] && !s[i]){
12 printf("\t%d->%d",v,i);
13 dfs(i);
14 }
15 }
16 }
17 int main()
18 {
19 int i,j,count=0;
20 printf("Enter the number of vertices:");
21 scanf("%d",&n);
22 for(i=1;i<=n;i++){
23 for(j=1;j<=n;j++)
24 a[i][j]=0;
25 s[i]=0;
26 }
27 printf("Enter the adjacency matrix:");
28 for(i=1;i<=n;i++){
29 for(j=1;j<=n;j++)
30 scanf("%d",&a[i][j]);
31 }
32 dfs(1);
33 printf("\n");
34 for(i=1;i<=n;i++){
Logs & others
Code::Blocks
Search results
Cccc
Build log
Build messages
CppCheck/Vera++
CppCheck/Vera++ messages
Cscope
Debugger
DoxyBlocks
Fortran info


depth.c [dfs] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main() : int
Management
Projects Files FSy
Workspace
hashing
dfs
hash.c
depth.c
printf("\t%d->%d",v,i);
dfs(i);
}
}
}
int main()
{
int i,j,count=0;
printf("Enter the number of vertices:");
scanf("%d",&n);
for(i=1;i<=n;i++){
for(j=1;j<=n;j++)
a[i][j]=0;
s[i]=0;
}
printf("Enter the adjacency matrix:");
for(i=1;i<=n;i++){
for(j=1;j<=n;j++)
scanf("%d",&a[i][j]);
}
dfs(1);
printf("\n");
for(i=1;i<=n;i++){
if(s[i])
count++;
}
if(count==n)
printf("Graph is connected");
else
printf("graph is not connected");
return 0;
}
Logs & others
Code::Blocks
Search results
Cccc
Build log
Build messages
CppCheck/Vera++
CppCheck/Vera++ messages
Cscope
Debugger
DoxyBlocks
Fortran info


C:\1BM23CS245\dfs\bin\Debu
Enter the number of vertices:4
Enter the adjacency matrix:
0 1 0 1
1 0 1 0
0 1 0 1
1 0 1 0
1->2 2->3 3->4
Graph is connected
Process returned 0 (0x0) execution time : 35.421 s
Press any key to continue.