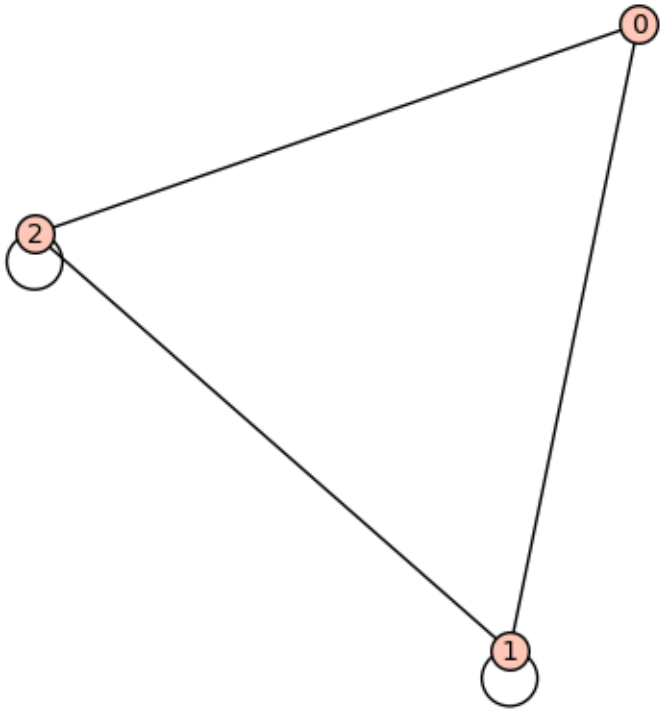
Best known constructions for s_ij

April 29, 2025

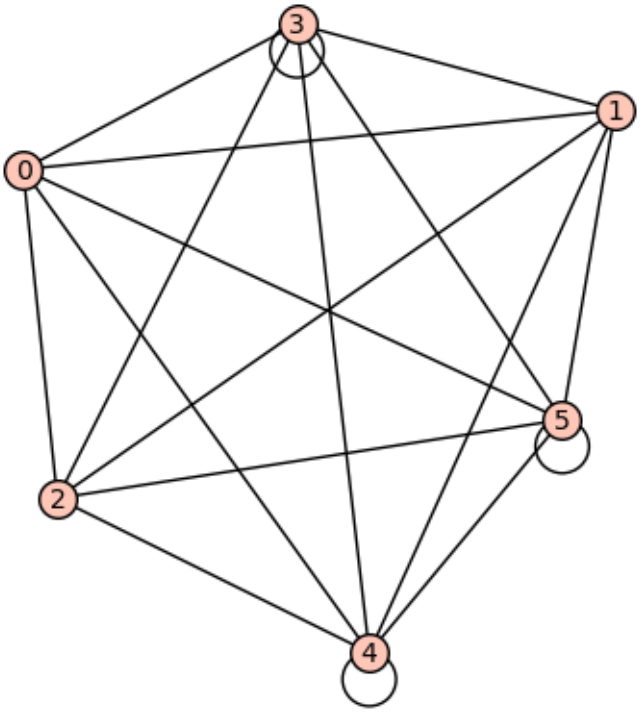
[1]:
```
def upper_bound_ij(i,j):
    return (1/2)*sqrt((i+j+1)/(i*(j+1)))

def upper_bound_0j(j):
    return (1/2)*(1+sqrt((j+2)/(j+1)))

def extremal_graph_ij(s, i, j):
    G = Graph(s)
    n = G.order()
    sp = G.spectrum()

    G.show()
    print(G.sparse6_string())
    print(f'lower s({i},{j}) = {((sp[i] - sp[n - j - 1])/n).n()}')
    if i == j == 0:
        print(f'upper s({i},{j}) = {(2/sqrt(3)).n()}')
    elif i == 0:
        print(f'upper s({i},{j}) = {upper_bound_0j(j).n()}')
    else:
        print(f'upper s({i},{j}) = {upper_bound_ij(i,j).n()}')
    return None
```

[2]:
```
# Best for s(0,0)
i,j = 0,0
K = graphs.CompleteGraph(3)
G = Graph(loops=True)
for e in K.edges():
    G.add_edge(e)
G.add_edge(1,1)
G.add_edge(2,2)
s = G.sparse6_string()
extremal_graph_ij(s,i,j)
```

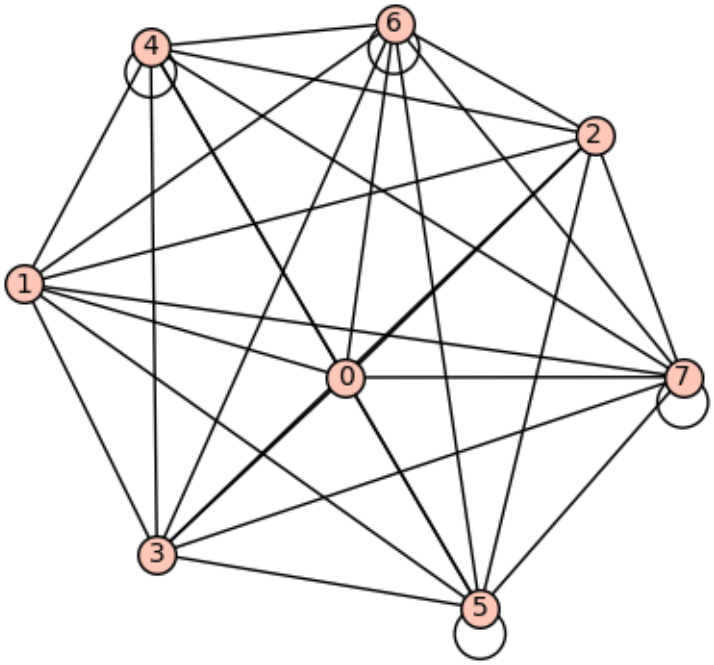



```
:B``V
lower s(0,0) = 1.15470053837925
upper s(0,0) = 1.15470053837925
```

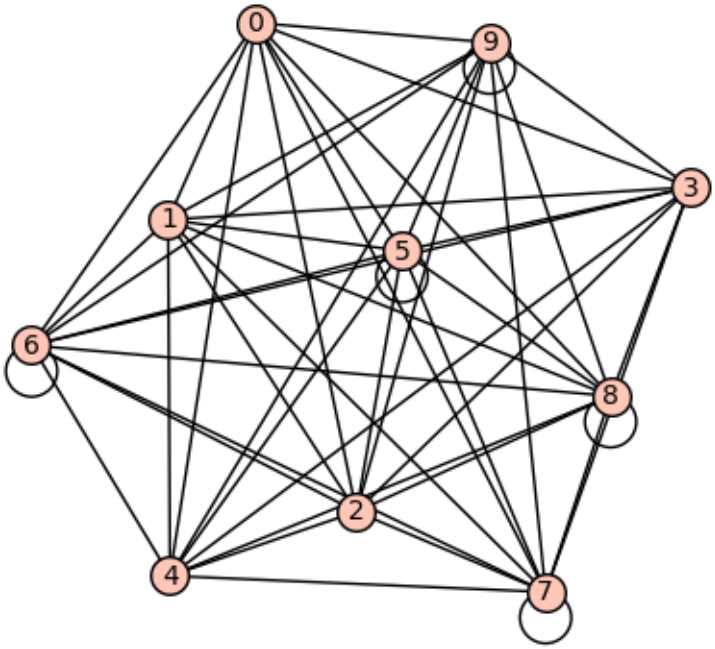
[3]:
```
# Best for s(0,j)
i = 0
for j in range(1,5):
    K = graphs.CompleteGraph(2*j+4)
    G = Graph(loops=True)
    for e in K.edges():
        G.add_edge(e)
    for v in range(j+2,2*j+4):
        G.add_edge(v,v)
    s = G.sparse6_string()
    extremal_graph_ij(s,i,j)
```



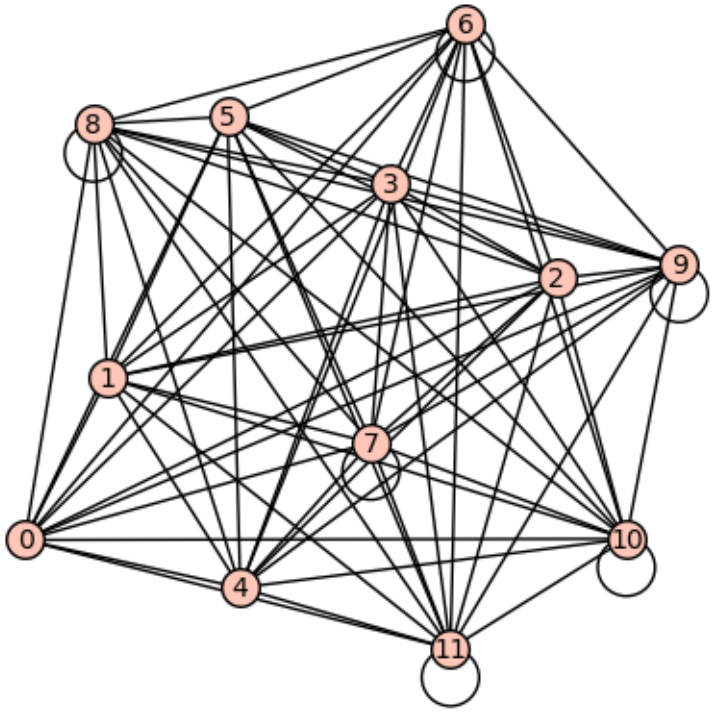

```
:Ea@_QM@Gs_QLD
lower s(0,1) = 1.09023021085819
upper s(0,1) = 1.11237243569579
```



```
:Ga@_Q_QLGCbPWCbPU_QLDX~
lower s(0,2) = 1.06639110926866
upper s(0,2) = 1.07735026918963
```

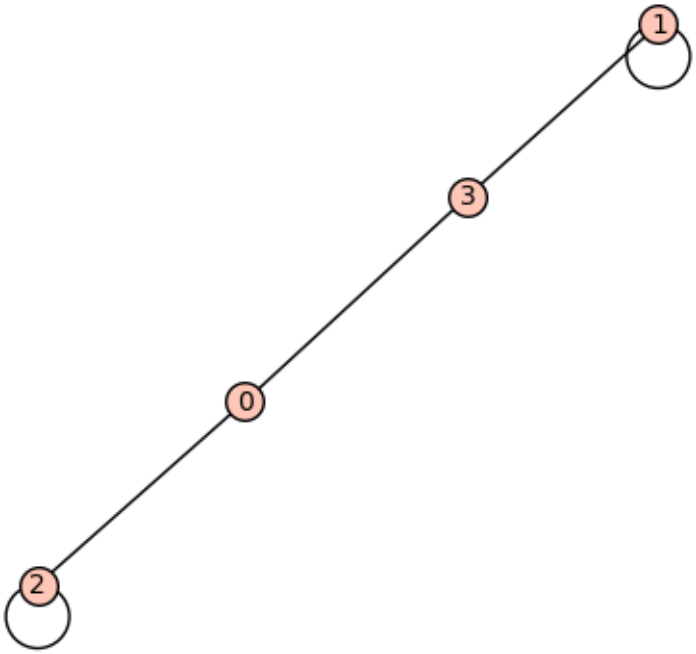



```
:I`?K?a_COw@CK`W@CK`ROAGXAeN?G`cIWyG@CK`RFOf
lower s(0,3) = 1.05249378105604
upper s(0,3) = 1.05901699437495
```

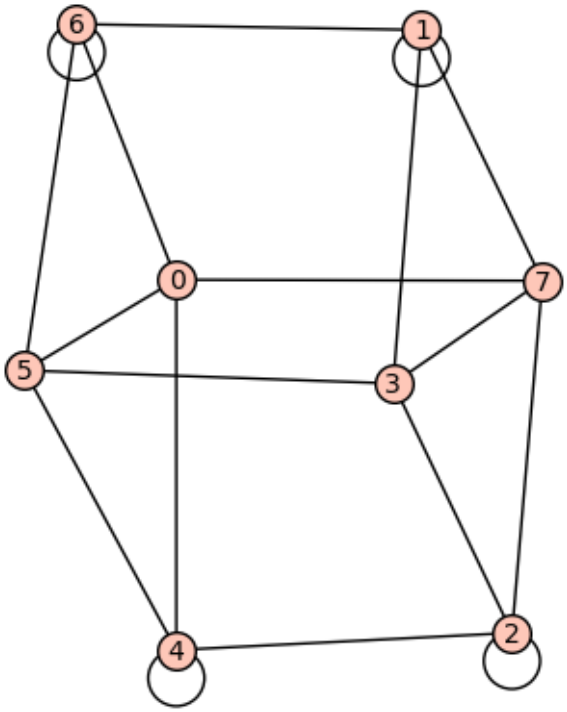



```
:K`?K?a_COw@CKc?aEOhg@CK`RF_COqDK\C?aEOhbgR?G`cIWyCi_COqDK\ATJ
lower s(0,4) = 1.04339977411635
upper s(0,4) = 1.04772255750517
```

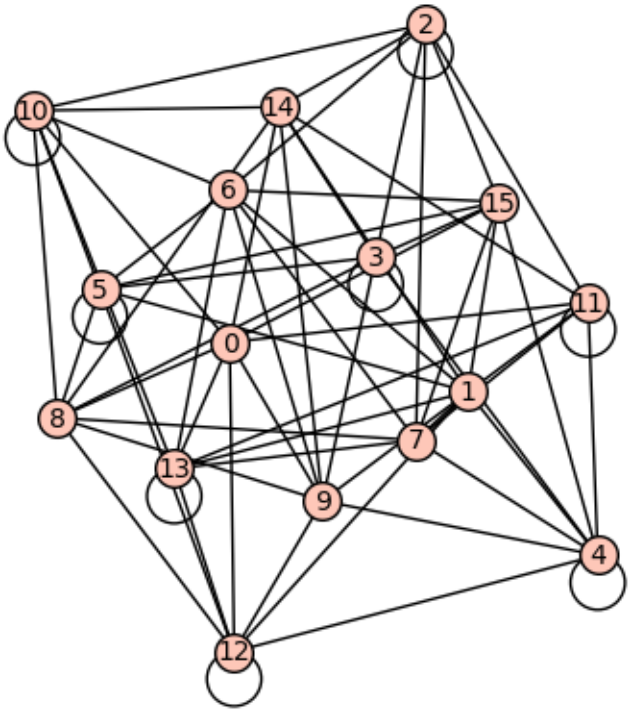
[4]:
```
# Best for s(k,k-1) if (2k)-Hadamard matrix exists
for k in range(1,5):
    i,j = k,k-1
    K = matrix([[1,-1],[-1,1]])
    try:
        H = hadamard_matrix(2*k)
    except:
        continue
    J = matrix.ones(ZZ,4*k,4*k)
    A = (1/2)*(K.tensor_product(H) + J)
    G = Graph(A)
    G.relabel(G.canonical_label())
    s = G.sparse6_string()
    extremal_graph_ij(s,i,j)
```




```
:CkSN
lower s(1,0) = 0.707106781186547
upper s(1,0) = 0.707106781186548
```



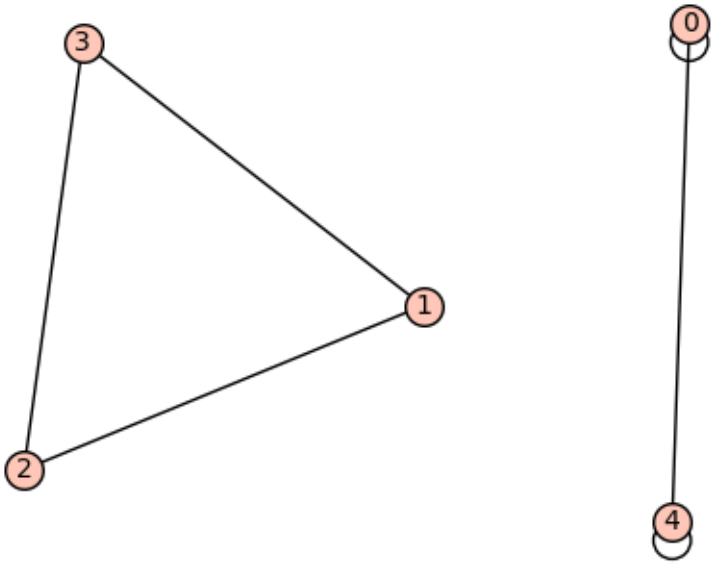
```
:GehIAQBQ@TgCb
lower s(2,1) = 0.500000000000000
upper s(2,1) = 0.500000000000000
```



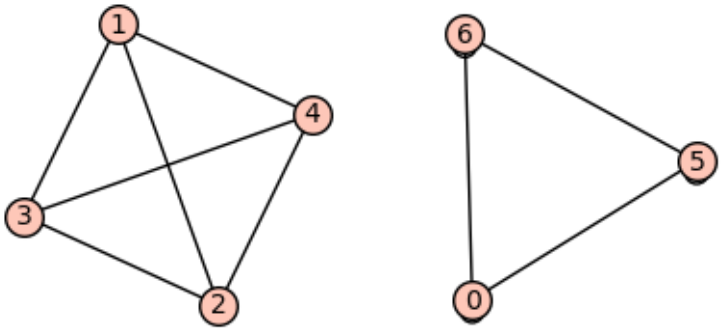

```
:OcIOwbHCXWaJCPBOES{@cKa?aeOi?aFQm?QDOdc?eMhZEoAGYCiV?G`cIW~
lower s(4,3) = 0.353553390593274
upper s(4,3) = 0.353553390593274
```

[5]:
```python
# Best for s(1,j)
i = 1
for j in range(1,5):
    K1 = graphs.CompleteGraph(j+2)
    K2 = graphs.CompleteGraph(j+1)
    G1 = Graph(loops=True)
    G2 = Graph(loops=True)
    for e in K1.edges():
        G1.add_edge(e)
    for e in K2.edges():
        G2.add_edge(e)
    for v in range(j+1):
        G2.add_edge(v,v)
```

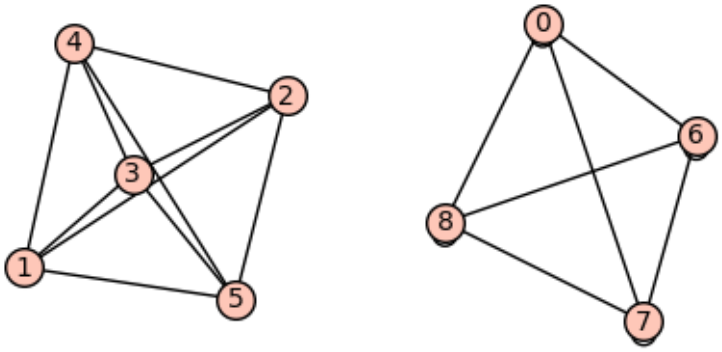
```
    G = G1.disjoint_union(G2)
    G.relabel(G.canonical_label())
    s = G.sparse6_string()
    extremal_graph_ij(s,i,j)
```



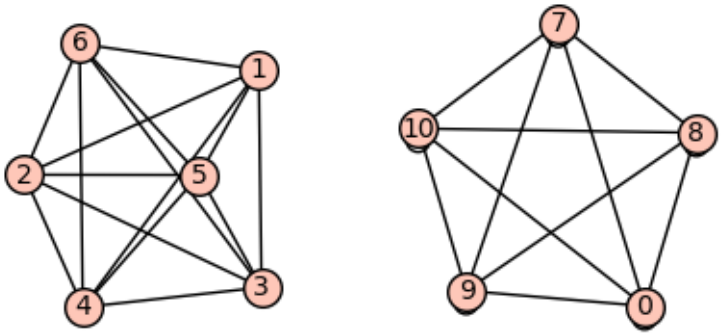

```
:DA`cgR
lower s(1,1) = 0.600000000000000
upper s(1,1) = 0.612372435695794
```




```
:FA`chGwUDZ
lower s(1,2) = 0.571428571428571
upper s(1,2) = 0.577350269189626
```

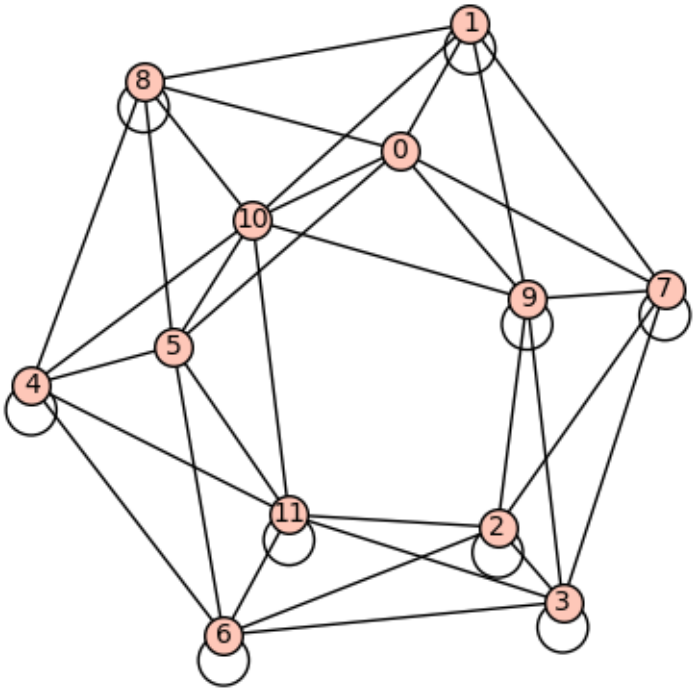



```
:H@GKPPCMG`c_Y@boK\F
lower s(1,3) = 0.555555555555556
upper s(1,3) = 0.559016994374947
```



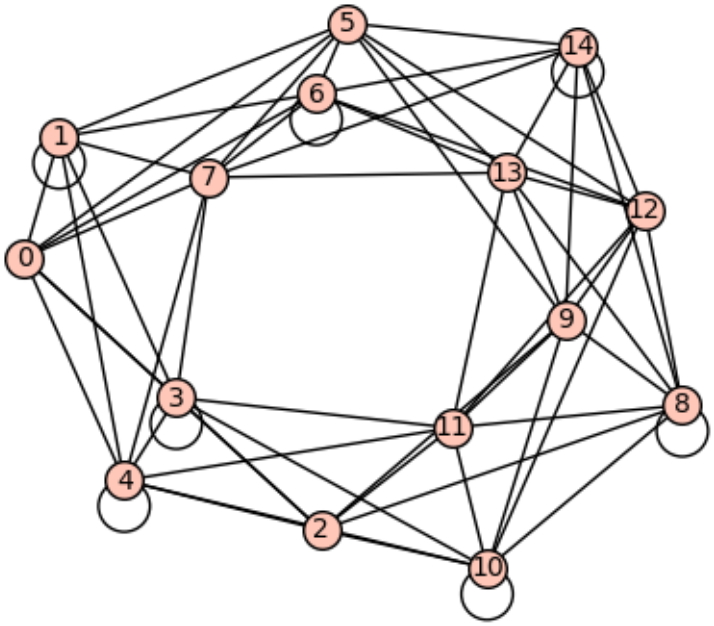

```
:J@GKPPCMG`caGXAoN?yGFOe@sHT
lower s(1,4) = 0.545454545454545
upper s(1,4) = 0.547722557505166
```

[6]:
```
# Best for s(2,2) named G_1
i,j = 2,2
s = ':K_ES`s_QOqDL?G`f_C`SOAGXsoAOiCqEOhdJ'
extremal_graph_ij(s,i,j)
```



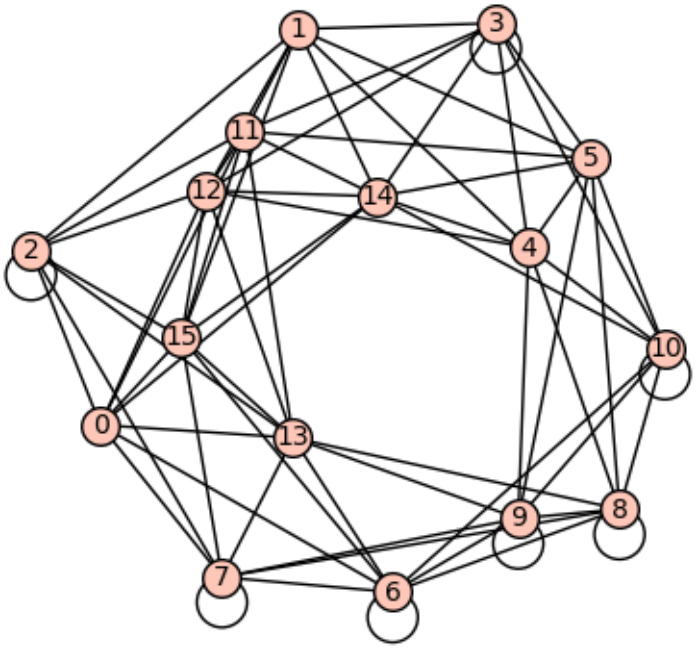

```
:K_ES`s_QOqDL?G`f_C`SOAGXsoAOiCqEOhdJ
lower s(2,2) = 0.440958551844098
upper s(2,2) = 0.456435464587638
```

[7]:
```
#Best for s(2,3) named G_2
i,j = 2,3
s = ':N_EC?aF?G`c_E?Qe_CXAecaPSQEPATQEPATTK`IdtK\\ATkiWyCkYz'
extremal_graph_ij(s,i,j)
```



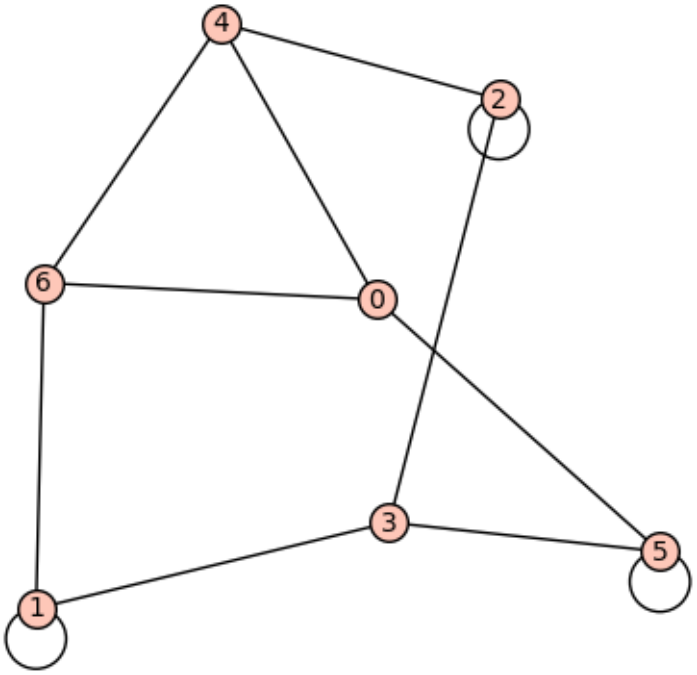

```
:N_EC?aF?G`c_E?Qe_CXAecaPSQEPATQEPATTK`IdtK\ATkiWyCkYz
lower s(2,3) = 0.414584287830377
upper s(2,3) = 0.433012701892219
```

[8]:
```
#Best for s(2,4) named G_3
i,j = 2,4
s = ':Oc?GgbaMGqOL?PbsIWyIDK\\AXcIXATOAGXW@CKawAK\\ATk_CXAiUq?PEMlbV^'
extremal_graph_ij(s,i,j)
```



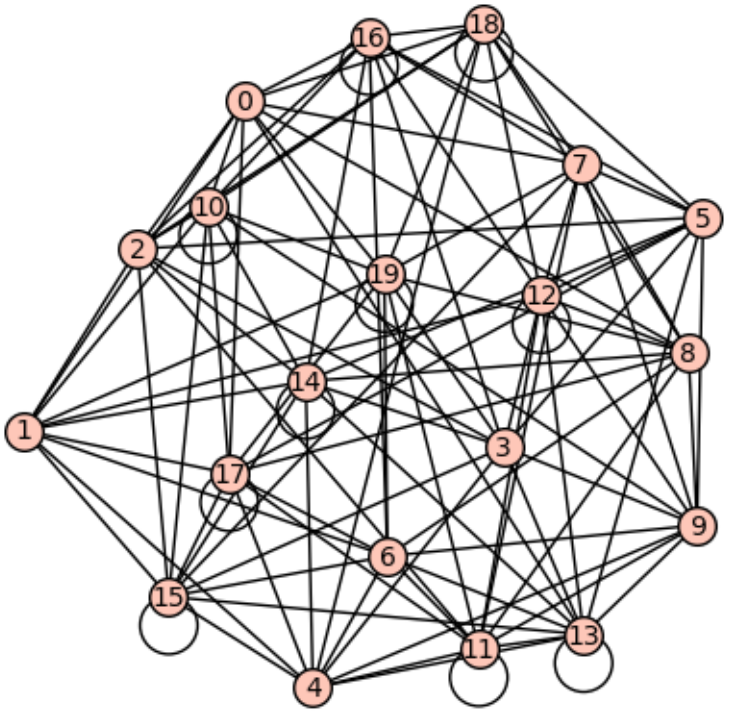

```
:Oc?GgbaMGqOL?PbsIWyIDK\AXcIXATOAGXW@CKawAK\ATk_CXAiUq?PEMlbV^
lower s(2,4) = 0.4042066628103555
upper s(2,4) = 0.4183300132670 38
```

[9]:
```
# Best for s(3,2) named G_4
i,j = 3,2
s = ':FehIA_t_S'
extremal_graph_ij(s,i,j)
```



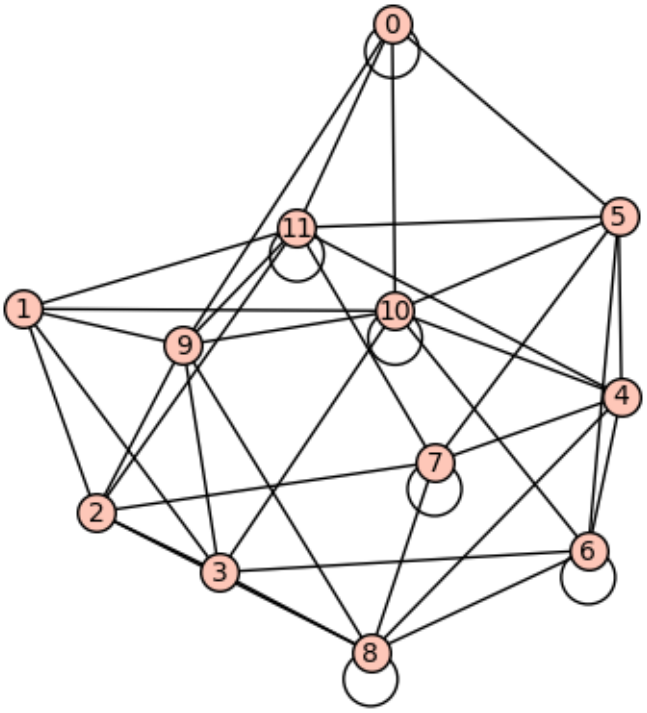

```
:FehIA_t_S
lower s(3,2) = 0.404061017820884
upper s(3,2) = 0.408248290463863
```

[10]:
```
#Best for s(3,3) named G_5
i,j = 3,3
s = ':
 ↪S__@a`BaB`C_D_EFbCDEFG_@AHIaCEFGHJ`BDFGHJK_BCDEHJKL`ABCDGILM`ABCEFILMN_ABDEFIKMO_@CDEGIJNP_
extremal_graph_ij(s,i,j)
```



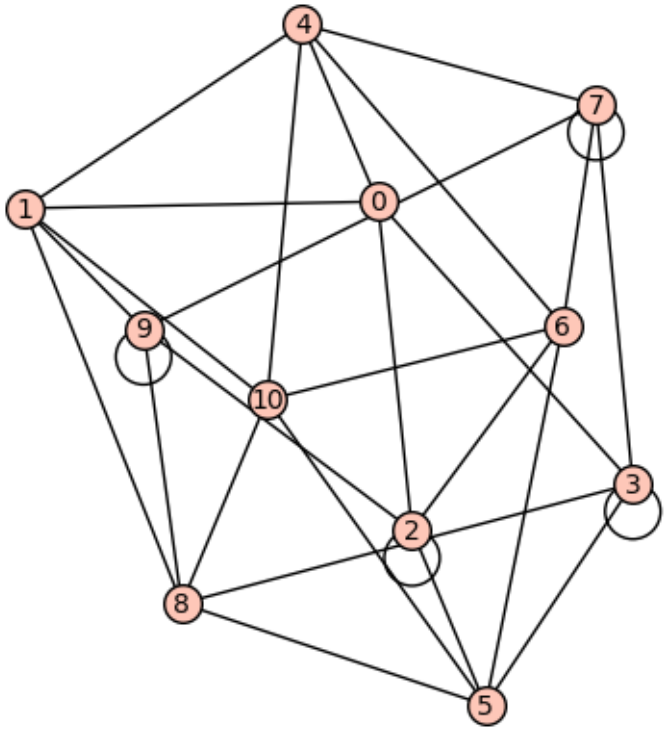

```
:S__@a`BaB`C_D_EFbCDEFG_@AHIaCEFGHJ`BDFGHJK_BCDEHJKL`ABCDGILM`ABCEFILMN_ABDEFIKM
O_@CDEGIJNP_ACDFGIKOQ_@BEFGIJPQR
lower s(3,3) = 0.368220060753790
upper s(3,3) = 0.381881307912987
```

[11]:
```
#Best for s(3,4) named G_6
i,j = 3,4
s = ':K@GKPT?QXAecOhxBGWyG@CLC?bGSqTOAG`RhV'
extremal_graph_ij(s,i,j)
```



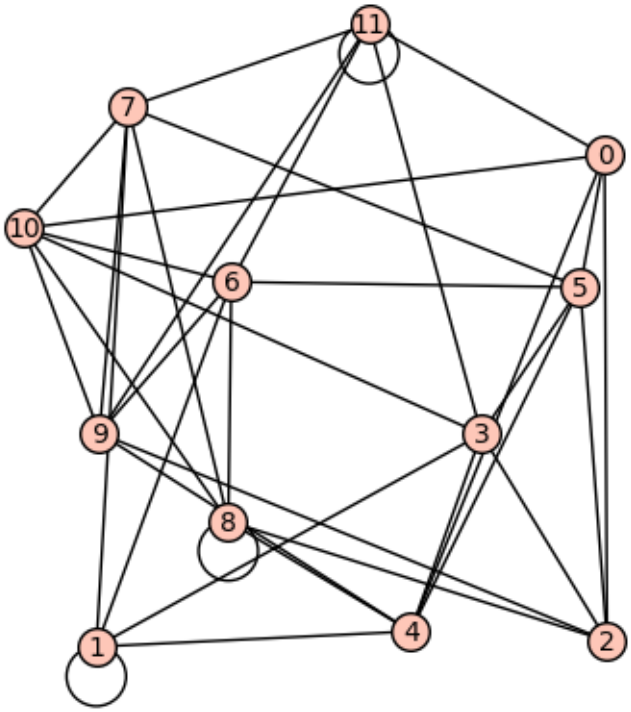

```
:K@GKPT?QXAecOhxBGWyG@CLC?bGSqTOAG`RhV
lower s(3,4) = 0.340561357881856
upper s(3,4) = 0.365148371670111
```

[12]:
```
#Best for s(4,4) named G_7
i,j = 4,4
s = ':J`?S@oBG[aDeOpwbJCPsHaOhc^'
extremal_graph_ij(s,i,j)
```



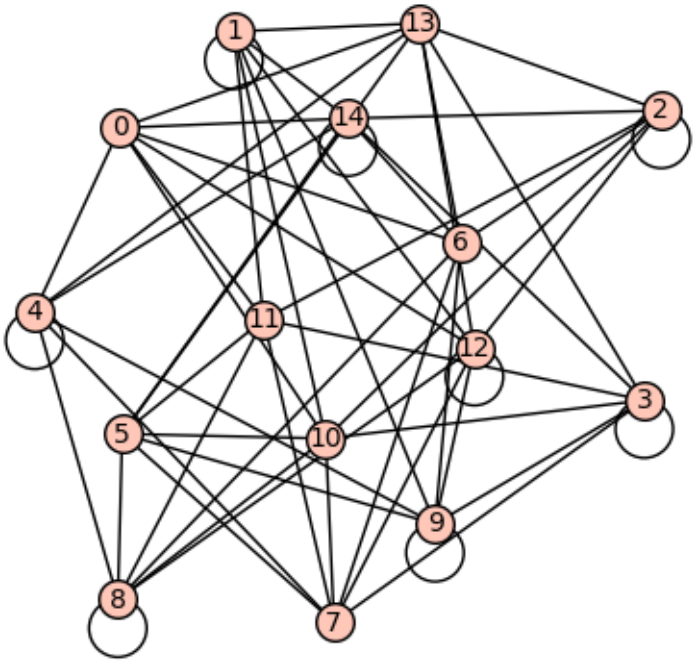

```
:J`?S@oBG[aDeOpwbJCPsHaOhc^
lower s(4,4) = 0.314918328648887
upper s(4,4) = 0.335410196624968
```

[13]:
```
#Best for s(3,1) named G_1^c
i = 3
j = 1
H = Graph(':K_ES`s_QOqDL?G`f_C`SOAGXsoAOiCqEOhdJ')
A = H.adjacency_matrix()
B = matrix.ones(ZZ, H.order(),H.order()) - A
G = Graph(B)
G.relabel(G.canonical_label())
s = G.sparse6_string()
extremal_graph_ij(s,i,j)
```



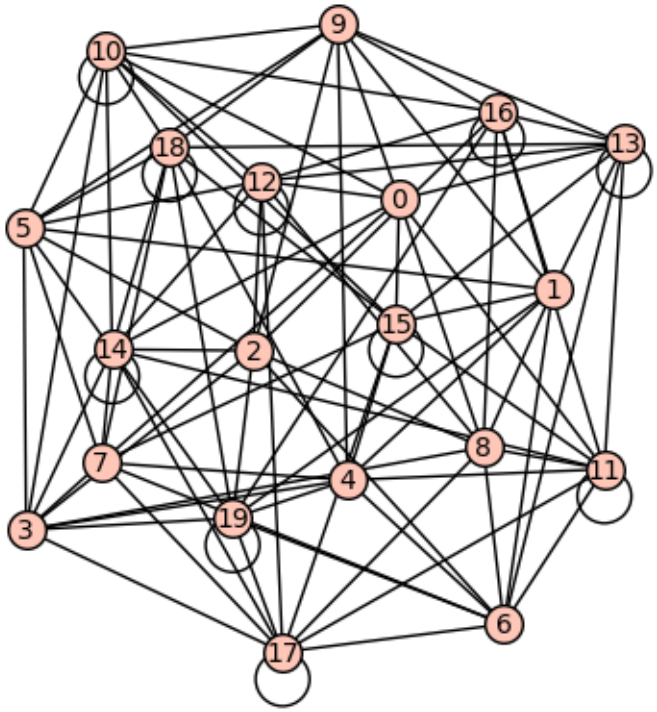

```
:KbAGg@F?OqPJCkaEMaPBF_KpsH_Kpsj
lower s(3,1) = 0.440958551844098
upper s(3,1) = 0.456435464587638
```

[14]:
```
#Best for s(4,1) named G_2^c
i = 4
j = 1
H = Graph(':N_EC?aF?G`c_E?Qe_CXAecaPSQEPATQEPATTK`IdtK\\ATkiWyCkYz')
A = H.adjacency_matrix()
B = matrix.ones(ZZ, H.order(),H.order()) - A
G = Graph(B)
G.relabel(G.canonical_label())
s = G.sparse6_string()
extremal_graph_ij(s,i,j)
```




```
:NbI[AU?IXAegSqGbGSqW@CKhsOAGXRg_CPsHX?G`cIXc?aEOhf^
lower s(4,1) = 0.414589061037904
upper s(4,1) = 0.433012701892219
```

[15]:
```
#Best for s(4,2) named G_5^c
i = 4
j = 2
H = Graph(':
 ↪S__@a`BaB`C_D_EFbCDEFG_@AHIaCEFGHJ`BDFGHJK_BCDEHJKL`ABCDGILM`ABCEFILMN_ABDEFIKMO_@CDEGIJNP_
A = H.adjacency_matrix()
B = matrix.ones(ZZ, H.order(),H.order()) - A
G = Graph(B)
G.relabel(G.canonical_label())
s = G.sparse6_string()
extremal_graph_ij(s,i,j)
```




```
:SbA`B`AB`AC_BCDE_@ABE_@ACD_BDHI_CEGJ_ADGIK_@EHJKL_ABFIJKM_@CFIJLN`AGHIJKLObEFGJ
KMNPcDFHILMNQ`ABCDEOPQR
lower s(4,2) = 0.367814729899738
upper s(4,2) = 0.381881307912987
```