The hat problem on a union of disjoint graphs

Abstract. The topic is the hat problem in which each of \( n \) players is randomly fitted with a blue or red hat. Then everybody can try to guess simultaneously his own hat color by looking at the hat colors of the other players. The team wins if at least one player guesses his hat color correctly, and no one guesses his hat color wrong; otherwise the team loses. The aim is to maximize the probability of winning. In this version every player can see everybody excluding himself. We consider such a problem on a graph, where vertices correspond to players, and a player can see each player to whom he is connected by an edge. The solution of the hat problem is known for cycles and bipartite graphs. We investigate the problem on a union of disjoint graphs.

2000 Mathematics Subject Classification: 05C99, 91A12.

Key words and phrases: Hat problem, Graph, Disjoint, Union.

1. Introduction. In the hat problem, a team of \( n \) players enters a room and a blue or red hat is randomly placed on the head of each player. Each player can see the hats of all of the other players but not his own. No communication of any sort is allowed, except for an initial strategy session before the game begins. Once they have had a chance to look at the other hats, each player must simultaneously guess the color of his own hat or pass. The team wins if at least one player guesses his hat color correctly and no one guesses his hat color wrong; otherwise the team loses. The aim is to maximize the probability of winning.

The hat problem with seven players, called the “seven prisoners puzzle”, was formulated by T. Ebert in his Ph.D. Thesis [12]. The hat problem was also the subject of articles in The New York Times [28], Die Zeit [6], and abcNews [27]. It was also one of the Berkeley Riddles [4].

The hat problem with \( 2^k - 1 \) players was solved in [14], and for \( 2^k \) players in [11]. The problem with \( n \) players was investigated in [7]. The hat problem and Hamming codes were the subject of [8]. The generalized hat problem with \( n \) people and \( q \) colors was investigated in [26].
There are many known variations of the hat problem (for a comprehensive list, see [24]). For example in [20] there was considered a variation in which players do not have to guess their hat colors simultaneously. In the papers [1, 10, 19] there was considered a variation in which passing is not allowed, thus everybody has to guess his hat color. The aim is to maximize the number of correct guesses. The authors of [17] investigated several variations of the hat problem in which the aim is to design a strategy guaranteeing a desired number of correct guesses. In [18] there was considered a variation in which the probabilities of getting hats of each colors do not have to be equal. The authors of [2] investigated a problem similar to the hat problem. There are \( n \) players which have random bits on foreheads, and they have to vote on the parity of the \( n \) bits.

The hat problem and its variations have many applications and connections to different areas of science (for a survey on this topic, see [24]), for example: information technology [5], linear programming [17], genetic programming [9], economics [1, 19], biology [18], approximating Boolean functions [2], and autoreducibility of random sequences [3, 12–15]. Therefore, it is hoped that the hat problem on a graph considered in this paper is worth exploring as a natural generalization, and may also have many applications.

We consider the hat problem on a graph, where vertices correspond to players and a player can see each player to whom he is connected by an edge. This variation of the hat problem was first considered in [21]. There were proven some general theorems about the hat problem on a graph, and the problem was solved on trees. Additionally, there was considered the hat problem on a graph such that the only known information are degrees of vertices. In [23] the hat problem was solved on the cycle \( C_4 \). In [25] the problem was solved on cycles on at least nine vertices. Then the problem was solved on all odd cycles [22]. Uriel Feige [16] conjectured that for any graph the maximum chance of success in the hat problem is equal to the maximum chance of success for the hat problem on the maximum clique in the graph. He provided several results that support this conjecture, and solved the hat problem for bipartite graphs and planar graphs containing a triangle. He also proved that the hat number of a union of disjoint graphs is the maximum hat number among that graphs.

In this paper we consider the hat problem on a union of disjoint graphs. By the union of two strategies (each for another graph) we mean the strategy for the union of that graphs such that every vertex behaves in the same way as in the proper strategy which is an element of the union. First, we give a sufficient condition for that the union of strategies gives worse chance of success than some component of the union. Next, we characterize when the union of strategies gives at least the same (better, the same, respectively) chance of success as each component of the union. Finally, we prove that there exists a disconnected graph for which there exists an optimal strategy such that every vertex guesses its color.

2. Preliminaries. For a graph \( G \), the set of vertices and the set of edges we denote by \( V(G) \) and \( E(G) \), respectively. If \( H \) is a subgraph of \( G \), then we write \( H \subseteq G \). Let \( v \in V(G) \). The degree of vertex \( v \), that is, the number of its neighbors, we denote by \( d_G(v) \). The path (complete graph, respectively) on \( n \) vertices we denote
by \( P_n \) (\( K_n \), respectively).

Let \( V(G) = \{v_1, v_2, \ldots, v_n\} \). By \( Sc = \{1, 2\} \) we denote the set of colors, where 1 corresponds to blue, and 2 corresponds to red.

By a case for a graph \( G \) we mean a function \( c : V(G) \to \{1, 2\} \), where \( c(v_i) \) means color of vertex \( v_i \). The set of all cases for the graph \( G \) we denote by \( C(G) \), of course \( |C(G)| = 2^{|V(G)|} \).

By a situation of a vertex \( v_i \) we mean a function \( s_i : V(G) \to Sc \cup \{0\} = \{0, 1, 2\} \), where \( s_i(v_j) \in Sc \) if \( v_i \) and \( v_j \) are adjacent, and 0 otherwise. The set of all possible situations of \( v_i \) in the graph \( G \) we denote by \( St_i(G) \), of course \( |St_i(G)| = 2^{d_G(v_i)} \).

We say that a case \( c \) for the graph \( G \) corresponds to a situation \( s_i \) of vertex \( v_i \) if \( c(v_j) = s_i(v_j) \), for every \( v_j \) adjacent to \( v_i \). This implies that a case corresponds to a situation of \( v_i \) if every vertex adjacent to \( v_i \) in that case has the same color as in that situation. Of course, to every situation of the vertex \( v_i \) correspond exactly \( 2^{|V(G)|-d_G(v_i)} \) cases.

By a guessing instruction of a vertex \( v_i \in V(G) \) we mean a function \( g_i : St_i(G) \to Sc \cup \{0\} = \{0, 1, 2\} \), which for a given situation gives the color \( v_i \) guesses it is, or 0 if \( v_i \) passes. Thus, a guessing instruction is a rule determining the behavior of a vertex in every situation.

Let \( c \) be a case, and let \( s_i \) be the situation (of vertex \( v_i \)) corresponding to that case. The guess of \( v_j \), in the case \( c \) is correct (wrong, respectively) if \( g_i(s_i) = c(v_i) \) (0 \( \neq \) \( g_i(s_i) \neq c(v_i) \), respectively). By result of the case \( c \) we mean a win if at least one vertex guesses its color correctly, and no vertex guesses its color wrong, that is, \( g_i(s_i) = c(v_i) \) (for some \( i \)) and there is no \( j \) such that 0 \( \neq \) \( g_j(s_j) \neq c(v_j) \). Otherwise the result of the case \( c \) is a loss.

By a strategy for the graph \( G \) we mean a sequence \( (g_1, g_2, \ldots, g_n) \), where \( g_i \) is the guessing instruction of vertex \( v_i \). The family of all strategies for a graph \( G \) we denote by \( \mathcal{F}(G) \).

If \( S \in \mathcal{F}(G) \), then the set of cases for the graph \( G \) for which the team wins (loses, respectively) using the strategy \( S \) we denote by \( W(S) \) (\( L(S) \), respectively). The set of cases for which the team loses, and some vertex guesses its color (no vertex guesses its color, respectively) we denote by \( Ls(S) \) (\( Ln(S) \), respectively). By the chance of success of the strategy \( S \) we mean the number \( p(S) = |W(S)|/|C(G)| \). By the hat number of the graph \( G \) we mean the number \( h(G) = \max\{p(S) : S \in \mathcal{F}(G)\} \). We say that a strategy \( S \) is optimal for the graph \( G \) if \( p(S) = h(G) \). The family of all optimal strategies for the graph \( G \) we denote by \( \mathcal{F}^0(G) \).

By solving the hat problem on a graph \( G \) we mean finding the number \( h(G) \).

Since for every graph we can apply a strategy in which one vertex always guesses it has, let us say, the first color, and the other vertices always pass, we immediately get the following lower bound on the hat number of a graph.

**Fact 2.1** For every graph \( G \) we have \( h(G) \geq 1/2 \).

The next solution of the hat problem on paths is a result from [21].

**Theorem 2.2** For every path \( P_n \) we have \( h(P_n) = 1/2 \).
3. Results. Let $G$ and $H$ be vertex-disjoint graphs, and let $S_1 \in \mathcal{F}(G)$ and $S_2 \in \mathcal{F}(H)$. By the union of the strategies $S_1$ and $S_2$ we mean the strategy $S \in \mathcal{F}(G \cup H)$ such that every vertex of $G$ behaves in the same way as in $S_1$, and every vertex of $H$ behaves in the same way as in $S_2$. If $S$ is the union of $S_1$ and $S_2$, then we write $S = S_1 \cup S_2$.

From now writing that $G$ and $H$ are graphs, we assume that they are vertex-disjoint.

In the following theorem we give a sufficient condition for that the union of strategies gives worse chance of success than some component of the union.

**Theorem 3.1** Let $G$ and $H$ be graphs, and let $S = S_1 \cup S_2$, where $S_1 \in \mathcal{F}(G)$ and $S_2 \in \mathcal{F}(H)$. Assume that $p(S_1) > 0$ and $p(S_2) > 0$. If $|\text{Ln}(S_1)| \cdot |\text{Ln}(S_2)| < |\text{Ls}(S_1)| \cdot |\text{Ls}(S_2)|$, then $p(S) < \max\{p(S_1), p(S_2)\}$.

**Proof** First, let us observe that $|\text{Ls}(S_1)| > 0$, otherwise no vertex guesses its color, and therefore $|W(S_1)| = 0$. Consequently, $p(S_1) = 0$, a contradiction. Similarly we get $|\text{Ls}(S_2)| > 0$. Now let us consider the strategy $S = S_1 \cup S_2$ for the graph $G \cup H$. The team wins if at least one vertex guesses its color correctly, and no vertex guesses its color wrong, thus the team wins if:

(i) some vertex of $G$ guesses its color correctly and no vertex of $G$ guesses its color wrong, and some vertex of $H$ guesses its color correctly and no vertex of $H$ guesses its color wrong, or

(ii) some vertex of $G$ guesses its color correctly and no vertex of $G$ guesses its color wrong, and no vertex of $H$ guesses its color, or

(iii) no vertex of $G$ guesses its color, and some vertex of $H$ guesses its color correctly and no vertex of $H$ guesses its color wrong.

This implies that

$$|W(S)| = |W(S_1)||W(S_2)| + |W(S_1)||\text{Ln}(S_2)| + |\text{Ln}(S_1)||W(S_2)|.$$

Since $|C(G \cup H)| = |C(G)||C(H)|$, we get

$$p(S) = \frac{|W(S)|}{|C(G \cup H)|} = \frac{|W(S_1)||W(S_2)| + |W(S_1)||\text{Ln}(S_2)| + |\text{Ln}(S_1)||W(S_2)|}{|C(G)||C(H)|}.$$

We have

$$p(S) \geq \max\{p(S_1), p(S_2)\} \Leftrightarrow (p(S) \geq p(S_1) \text{ and } p(S) \geq p(S_2)).$$

Now we get the following chain of equivalences

$$p(S_1) \leq p(S) \Leftrightarrow \frac{|W(S_1)|}{|C(G)|} \leq \frac{|W(S_1)||W(S_2)| + |W(S_1)||\text{Ln}(S_2)| + |\text{Ln}(S_1)||W(S_2)|}{|C(G)||C(H)|}$$

$$\Leftrightarrow |W(S_1)||C(H)| \leq |W(S_1)||W(S_2)| + |W(S_1)||\text{Ln}(S_2)|$$

$$+ |\text{Ln}(S_1)||W(S_2)|$$

$$\Leftrightarrow |C(H)| \leq |W(S_2)| + |\text{Ln}(S_2)| + \frac{|\text{Ln}(S_1)||W(S_2)|}{|W(S_1)|}.$$
\[ \Ls(S_2) \leq \frac{|\Ln(S_1)||\W(S_1)|}{|\W(S_2)|}. \]

Similarly we get
\[ p(S) \geq p(S_2) \iff \frac{|\Ln(S_1)||\W(S_1)|}{|\W(S_2)|} \geq |\Ls(S_1)|. \]

Therefore we have
\[ p(S) \geq \max\{p(S_1), p(S_2)\} \]
if and only if
\[ \frac{|\Ln(S_1)||\W(S_2)|}{|\W(S_1)|} \geq |\Ls(S_2)| \quad \text{and} \quad \frac{|\Ln(S_2)||\W(S_1)|}{|\W(S_2)|} \geq |\Ls(S_1)|. \]

Consequently,
\[ p(S) \geq \max\{p(S_1), p(S_2)\} \Rightarrow |\Ln(S_1)| \cdot |\Ln(S_2)| \geq |\Ls(S_1)| \cdot |\Ls(S_2)|. \]

Equivalently,
\[ |\Ln(S_1)| \cdot |\Ln(S_2)| < |\Ls(S_1)| \cdot |\Ls(S_2)| \Rightarrow p(S) < \max\{p(S_1), p(S_2)\}. \]

**Corollary 3.2** Let \( G \) and \( H \) be graphs, and let \( S = S_1 \cup S_2 \), where \( S_1 \in \mathcal{F}(G) \) and \( S_2 \in \mathcal{F}(H) \). Assume that \( p(S_1) > 0 \) and \( p(S_2) > 0 \). If \( |\Ln(S_1)| = 0 \) or \( |\Ln(S_2)| = 0 \), then \( p(S) < \max\{p(S_1), p(S_2)\} \).

**Proof** As we have observed in the proof of Theorem 3.1, we have \( |\Ls(S_1)| > 0 \) and \( |\Ls(S_2)| > 0 \). Therefore \( \Ln(S_1) \cdot \Ln(S_2) = 0 < |\Ls(S_1)| \cdot |\Ls(S_2)| \). Now, by Theorem 3.1 we have \( p(S) < \max\{p(S_1), p(S_2)\} \).

From now writing \( S_1 \in \mathcal{F}(G) \) and \( S_2 \in \mathcal{F}(H) \), we assume that \( p(S_1) > 0 \), \( p(S_2) > 0 \), and \( |\Ln(S_1)| \cdot |\Ln(S_2)| \geq |\Ls(S_1)| \cdot |\Ls(S_2)| \).

The following theorem determines when the union of strategies gives at least the same chance of success as each component of the union.

**Theorem 3.3** If \( G \) and \( H \) are graphs and \( S = S_1 \cup S_2 \), where \( S_1 \in \mathcal{F}(G) \) and \( S_2 \in \mathcal{F}(H) \), then
\[ p(S) \geq \max\{p(S_1), p(S_2)\} \iff \frac{|\W(S_1)|}{|\W(S_2)|} \in \left[ \frac{|\Ln(S_1)|}{|\Ln(S_2)|}, \frac{|\Ln(S_1)|}{|\Ln(S_2)|} \right]. \]

**Proof** From the proof of Theorem 3.1 we know that
\[ p(S) \geq \max\{p(S_1), p(S_2)\} \]
if and only if
\[ \frac{|\Ln(S_1)||\W(S_2)|}{|\W(S_1)|} \geq |\Ls(S_2)| \quad \text{and} \quad \frac{|\Ln(S_2)||\W(S_1)|}{|\W(S_2)|} \geq |\Ls(S_1)|. \]
Since $|\text{Ln}(S_1)| > 0$, $|\text{Ln}(S_2)| > 0$, and $|\text{Ls}(S_2)| > 0$ (see the proof of Theorem 3.1), the condition above is equivalent to that

$$\frac{|W(S_1)|}{|W(S_2)|} < \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|} \quad \text{and} \quad \frac{|W(S_1)|}{|W(S_2)|} > \frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|},$$

that is

$$\frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|} < \frac{|W(S_1)|}{|W(S_2)|} < \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|}.$$

The interval

$$\left[ \frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|}, \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|} \right]$$

is nonempty, because $|\text{Ln}(S_1)| \cdot |\text{Ln}(S_2)| \geq |\text{Ls}(S_1)| \cdot |\text{Ls}(S_2)|$. Concluding,

$$p(S) > \max\{p(S_1), p(S_2)\} \iff \frac{|W(S_1)|}{|W(S_2)|} \in \left[ \frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|}, \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|} \right].$$

**Corollary 3.4** If $G$ and $H$ are graphs and $S = S_1 \cup S_2$, where $S_1 \in \mathcal{F}(G)$ and $S_2 \in \mathcal{F}(H)$, then

$$p(S) < \max\{p(S_1), p(S_2)\} \iff \frac{|W(S_1)|}{|W(S_2)|} \notin \left[ \frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|}, \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|} \right].$$

The following two theorems determine when the union of strategies gives better (or the same) chance of success than each component of the union. The proof of each one of these two theorems is similar to proofs of Theorems 3.1 and 3.3. Therefore we do not prove them.

**Theorem 3.5** If $G$ and $H$ are graphs and $S = S_1 \cup S_2$, where $S_1 \in \mathcal{F}(G)$ and $S_2 \in \mathcal{F}(H)$, then

$$p(S) > \max\{p(S_1), p(S_2)\} \iff \frac{|W(S_1)|}{|W(S_2)|} \in \left( \frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|}, \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|} \right).$$

**Theorem 3.6** If $G$ and $H$ are graphs and $S = S_1 \cup S_2$, where $S_1 \in \mathcal{F}(G)$ and $S_2 \in \mathcal{F}(H)$, then

$$p(S) = p(S_1) \iff \frac{|W(S_1)|}{|W(S_2)|} = \frac{|\text{Ln}(S_1)|}{|\text{Ls}(S_2)|}$$

and

$$p(S) = p(S_2) \iff \frac{|W(S_1)|}{|W(S_2)|} = \frac{|\text{Ls}(S_1)|}{|\text{Ln}(S_2)|}.$$
Corollary 3.7 Assume that $G$ and $H$ are graphs, and $S = S_1 \cup S_2$, where $S_1 \in \mathcal{F}(G)$ and $S_2 \in \mathcal{F}(H)$. Let $i \in \{1,2\}$ be such that $p(S_i) = \max\{p(S_1),p(S_2)\}$, and let $j \in \{1,2\}$, $j \neq i$. Then

$$p(S) = \max\{p(S_1),p(S_2)\} \Leftrightarrow \frac{|W(S_i)|}{|W(S_j)|} = \frac{|Ls(S_i)|}{|Ls(S_j)|}.$$  

It is possible to prove that there exists a disconnected graph for which there exists an optimal strategy such that every vertex guesses its color. First, we solve the hat problem on such graph, $K_2 \cup K_2$.

Fact 3.8 $h(K_2 \cup K_2) = 1/2$.

Proof We have $K_2 \cup K_2 \subseteq P_4$, thus $h(K_2 \cup K_2) \leq h(P_4)$. By Theorem 2.2 we have $h(P_4) = 1/2$. Therefore $h(K_2 \cup K_2) \leq 1/2$. On the other hand, by Fact 2.1 we have $h(K_2 \cup K_2) \geq 1/2$. $\blacksquare$

Fact 3.9 There exists an optimal strategy for the graph $K_2 \cup K_2$ such that every vertex guesses its color.

Proof Let $S' = (g_1, g_2) \in \mathcal{F}(K_2)$ be the strategy as follows.

$$g_1(s_1) = \begin{cases} 1 & \text{if } s_1(v_2) = 1, \\ 0 & \text{otherwise}; \end{cases}$$

$$g_2(s_2) = \begin{cases} 2 & \text{if } s_2(v_1) = 2, \\ 0 & \text{otherwise.} \end{cases}$$

It means that the vertices proceed as follows.

- **The vertex** $v_1$. If $v_2$ has the first color, then it guesses it has the first color, otherwise it passes.

- **The vertex** $v_2$. If $v_1$ has the second color, then it guesses it has the second color, otherwise it passes.

All cases we present in Table 1, where the symbol $+$ means correct guess (success), $-$ means wrong guess (loss), and blank square means passing.

<table>
<thead>
<tr>
<th>No</th>
<th>The color of $v_1$</th>
<th>The guess of $v_2$</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>+</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>+</td>
</tr>
</tbody>
</table>

Table 1
From Table 1 we know that $|W(S')| = 2$, $|Ln(S')| = 1$, and $|Ls(S')| = 1$. Since $|C(K_2)| = 4$, we get $p(S') = |W(S')|/|C(K_2)| = 2/4 = 1/2$. Let $S = S_1 \cup S_2 \in \mathcal{F}(K_2 \cup K_2)$, where $S_1 = S_2 = S'$. We have $|W(S_1)| = |W(S_2)| = |W(S')| = 1$ and $|Ln(S_1)|/|Ls(S_2)| = |Ln(S')|/|Ls(S')| = 1/1 = 1$. Since $|W(S_1)|/|W(S_2)| = |Ln(S_1)|/|Ls(S_2)|$, by Theorem 3.6 we get $p(S) = p(S_1)$. We have $S_1 = S'$ and $p(S') = 1/2$, thus $p(S) = 1/2$. By Fact 3.8 we have $h(K_2 \cup K_2) = 1/2$. This implies that the strategy $S$ is optimal. In this strategy every vertex guesses its color.

Now we prove this fact elementary.

PROOF Let $E(K_2 \cup K_2) = \{v_1v_2, v_3v_4\}$, and let $S = (g_1, g_2, g_3, g_4)$ be the strategy for $K_2 \cup K_2$ as follows.

$$
\begin{align*}
g_1(s_1) &= \begin{cases} 
1 & \text{if } s_1(v_2) = 1, \\
0 & \text{otherwise}; 
\end{cases} \\
g_2(s_2) &= \begin{cases} 
2 & \text{if } s_2(v_1) = 2, \\
0 & \text{otherwise}; 
\end{cases} \\
g_3(s_3) &= \begin{cases} 
1 & \text{if } s_3(v_4) = 1, \\
0 & \text{otherwise}; 
\end{cases} \\
g_4(s_4) &= \begin{cases} 
2 & \text{if } s_4(v_3) = 2, \\
0 & \text{otherwise}. 
\end{cases}
\end{align*}
$$

It means that the vertices proceed as follows.

- **The vertex** $v_1$. If $v_2$ has the first color, then it guesses it has the first color, otherwise it passes.

- **The vertex** $v_2$. If $v_1$ has the second color, then it guesses it has the second color, otherwise it passes.

- **The vertex** $v_3$. If $v_4$ has the first color, then it guesses it has the first color, otherwise it passes.

- **The vertex** $v_4$. If $v_3$ has the second color, then it guesses it has the second color, otherwise it passes.

All cases we present in Table 2. From this table we get $|W(S)| = 8$. We have $|C(K_2 \cup K_2)| = 16$, thus $p(S) = 8/16 = 1/2$. Similarly as in the previous proof we conclude that the strategy $S$ is optimal. In this strategy every vertex guesses its color.
Table 2

<table>
<thead>
<tr>
<th>No</th>
<th>The color of</th>
<th>The guess of</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$v_1$</td>
<td>$v_2$</td>
<td>$v_3$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

References


The hat problem on a union of disjoint graphs


Marcin Krzywkowski
Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology
Narutowicza 11/12, 80–233 Gdańsk, Poland
E-mail: marcin.krzywkowski@gmail.com

(Received: 1.01.2011)