begindocument

Problem Statement
Input: Nhu trong file text.
Variables: Nhu trong file text.
Invariants: Nhu trong file text.
Constraints: Nhu trong file text.
Output: Nhu trong file text.

Input: As problem statement
Output: As problem statement

```
1  foreach cc = 1,..., n do
2  |  F ← ∅;
3  |  course ← c(cc);
4  |  if TL(course) ≠ ∅ then
5  |  |  foreach t' ∈ TL(course) do
6  |  |  |  if isFeasibleToAssign(t', cc) = true then
7  |  |  |  |  F ← F ∪ {t'};
8  |  |  |  end
9  |  |  end
10 |  |  if F ≠ ∅ then
11 |  |  |  F1 ← {t ∈ F | pickScore(t, cc) is maximal};
12 |  |  |  t' ← random element of F1;
13 |  |  |  gv[cc] ← t';
14 |  |  end
15 |  end
16 end
17 return gv;
```

Algorithm 1: AssigningTeacher

Input: classcourse *cc*.
teacher *t*.
conflict matrix *cf*.

Output: Return *true* if it is possible to assign teacher t to classcourse *cc*, else return *false*.

```
1 foreach cc' ∈ AS(t) do
2 |   if cf(cc, cc') = true then
3 |   |   return false;
4 |   end
5 end
6 return true;
```

Algorithm 2: IsFeasibleToAssign(t,cc)