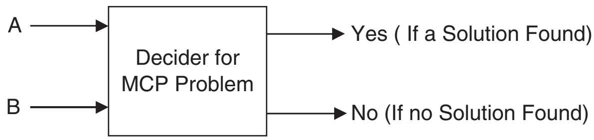
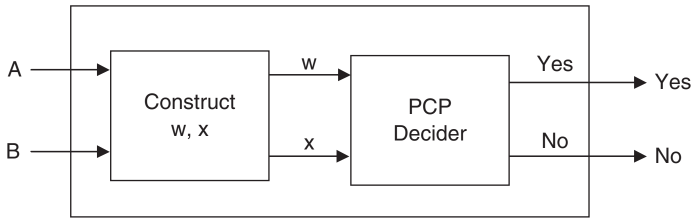
Let there be a decider for MPCP. The decider takes two inputs A and B and halts and accepts if a solution is found or halts and rejects if a solution is not found. The following is the decider for this.



The MPC problem with input (A, B) can be reduced to the input of a PCP decider (w, x) as described earlier.



The two set of strings w and x have a solution if and only if the two set of strings A and B have a solution. Since the MPC problem is undecidable, PCP is also undecidable.

Example 10.25 Prove that the problem 'whether an arbitrary CFG G is ambiguous' is undecidable.

Solution: To prove this, we reduce the PCP problem to the A_{Amb} problem, where A_{Amb} is considered as a decider for the ambiguity of the CFG problem. The PCP problem decider takes two lists A and B as input and generates a CFG G. If the PCP decider has a solution for the two lists A and B, then the CFG G is ambiguous.

This reduction is done by the following process.

Let $A = \{w_1, w_2, \ldots.. w_n\}$ and $B = \{x_1, x_2, \ldots.. x_n\}$.

Let $a_1, a_2 \ldots.. a_n$ be a list of new symbols generated by the languages.

$$L_A = \{w_{i1}\ w_{i2} \ldots..\ w_{im}\ a_{im}\ a_{im-1} \ldots..\ a_{i2}\ a_{i1},\ \text{where } 1 \le i \le n\ ;\ m \ge 1\}$$

and

$$L_B = \{x_{i1}\ x_{i2} \ldots..\ x_{im}\ a_{im}\ a_{im-1} \ldots..\ a_{i2}\ a_{i1},\quad \text{where } 1 \le i \le n\ ;\ m \ge 1\}$$

Both L_A and L_B can be constructed from the following grammar.

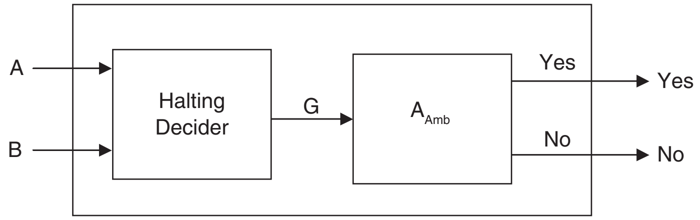
$$S_A \to w_iS_Aa_i/w_ia_i$$
$$S_B \to x_iS_Ba_i/x_ia_i$$

Both of them are CFG.

We can easily construct a new CFG by the union operation of the two CFG since CFG are closed under the union operation.

$$S \to S_A/S_B$$
$$S_A \to w_iS_Aa_i/w_ia_i$$
$$S_B \to x_iS_Ba_i/x_ia_i$$

If the given instance of the PCP has a solution for $i_1, i_2, \ldots\ldots i_m$, then the string generated in the top and bottom halves are $w_{i1}\ w_{i2} \ldots\ldots w_{im} = x_{i1}\ x_{i2} \ldots\ldots x_{im}$. Thus, $L_A = L_B$. For the grammar $L_A \cup L_B$, there exist two derivations of a same string, which means two parse trees. This makes $L_A \cup L_B$ ambiguous.



But it is already proved that the halting problem is undecidable. So, the problem A_{Amb} is also undecidable.

What We Have Learned So Far

1. A Turing-acceptable language is called Turing-recognizable or recursively enumerable.
2. A language is recursively enumerable if there is a Turing machine that halts and accepts the language or halts and rejects or loops for an infinite time.
3. A language is called Turing-decidable if there exists a Turing machine which halts and accepts the language. Otherwise, it is undecidable.
4. A recursive language is a subset of a recursively enumerable language.
5. A reduction is a process of converting one problem into another solved problem in such a way that the solution of the second problem can be used to solve the first problem.
6. The complement of a decidable language is also decidable.
7. The problem 'an arbitrary string w is accepted by an arbitrary Turing machine M' is known as the membership problem, which is an undecidable problem.
8. The problem 'a string w halts on a Turing machine M' is known as the halting problem, which is an undecidable problem.
9. Given two sequences of strings $w_1, w_2, \ldots, w_n$ and $x_1, x_2, \ldots, x_n$ over Σ. The solution of the problem is to find a non-empty sequence of integers $i_1, i_2, \ldots, i_k$ such that $w_{i1}w_{i2} \ldots w_{ik} = x_{i1}x_{i2} \ldots x_{ik}$ is known as the Post correspondence problem (PCP).
10. Given two sequences of strings $w_1, w_2, \ldots, w_n$ and $x_1, x_2, \ldots, x_n$ over Σ. The solution of the problem is to find a non-empty sequence of integers $i_2, i_3, \ldots, i_k$ such that $w_1w_{i2}w_{i3} \ldots w_{ik} = x_1x_{i2}x_{i3} \ldots x_{ik}$ is known as the modified Post correspondence problem (MPCP).
11. The PCP and MPCP are both undecidable problems.

Solved Problems

1. Find at least three solutions to the PCP defined by the dominoes.

	1	2	3
X:	1	10	10111
Y:	111	0	10

[UPTU 2002]

Solution: The solution is possible as X1, Y1 and X3, and Y3 have the same initial substring. The solution is 3112.

	3	1	1	2
X	10111	1	1	10
Y	10	111	111	0

Repeating the sequence 3112, we can get multiple solutions.

2. Does the PCP with two lists X = {b, bab^3, ba} and Y = {b^3, ba, a} have a solution.

[UPTU 2003]

Solution: The solution is possible as X1, Y1 and X2, and Y2 have the same initial substring. The solution is 2113.

	2	1	1	3
X	babbb	b	b	ba
Y	ba	bbb	bbb	a

3. Does the PCP with two list X = {(a, ba), (b^{2}a, a^3), (a^{2}b, ba)} has a solution?

Solution: The solution is not possible as none of the top and bottom parts of the list have the same starting character.

4. Show that the following PCP has a solution and give the solution.

	A	B
1	11	111
2	100	001
3	111	11

[JNTU 2008]

Solution: The solution is possible as the A and B parts of the first and third lists have the same starting characters. The solution is 123.

	1	2	3
X	11	100	111
Y	111	001	11

5. The lists A and B given in the following are an instance of PCP.

	A	B
1	0	01
2	0101	1
3	100	0010

Find the solution for the given PCP.

Solution: The solution is 1332.

	1	3	3	2
A:	0	100	100	0101
B:	01	0010	0010	1

6. Prove that if L_1 is regular and L_2 is context free, then $L_1 \cup L_2^c$ is recursive.

Solution: L_1 is regular, so it is recursive. The complement of the context-free language is not context free. But, the context-free language is recursive and the complement of the recursive language is recursive. It is proved that the union of two recursive languages is recursive. Hence, $L_1 \cup L^c_2$ is recursive.

Multiple Choice Questions

1. The Turing machine accepts
   a) Regular language
   b) Context-free language
   c) Context-sensitive language
   d) All of these
2. A language is recursively enumerable if a Turing machine
   a) Halts and accepts
   b) Halts and rejects
   c) loops forever
   d) performs (a), (b), or (c)
3. A language L is called decidable (or just decidable), if there exists a Turing machine M which
   a) Accepts L
   b) Rejects L
   c) Loops forever on L
   d) performs (a), (b), or (c)
4. Find the true statement
   a) A recursively enumerable language is a subset of a recursive language
   b) A recursive language is a subset of a recursively enumerable language
   c) Both are equivalent
   d) Both may loop forever on the input to a Turing machine
5. Which is false for recursive language?
   a) Union of two recursive languages is recursive