

Assignment E

Team: Andreea Matei & Florin Deleanu

Test Cases	3
Test 1	3
Test 2	5
Test 3	9
Test 4	12
Test 5	14

Test Cases

Test 1

Input

- *nfa.txt:*

```
(assert (and  
  
  (Accept "aa")  
  (Accept "ab")  
  (Accept "florin")  
  (Accept "f")  
  (Accept "ada")  
  (not (Accept ""))  
  (not (Accept "flori"))  
  (not (Accept "flo"))  
  (not (Accept "flor"))  
  (not (Accept "bba"))  
  (not (Accept "bbb"))  
  
))
```

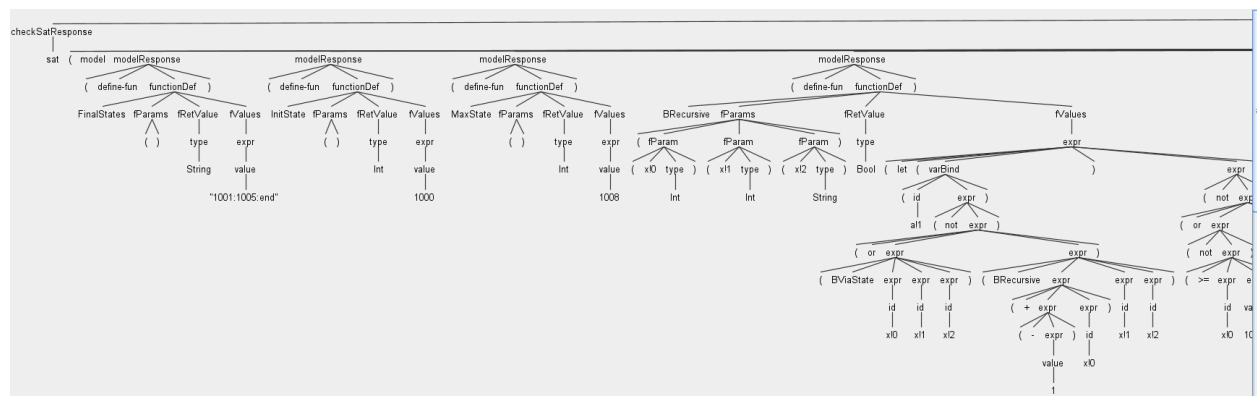
- *input.txt:*

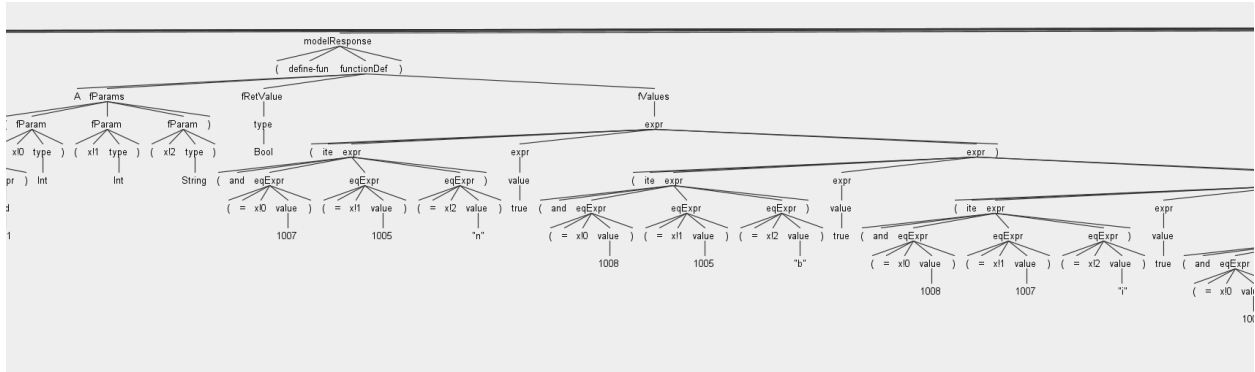
```

sat
(model
  (define-fun FinalStates () String
    "1001:1005:end")
  (define-fun InitState () Int
    1000)
  (define-fun MaxState () Int
    1008)
  (define-fun BRecursive ((x!0 Int) (x!1 Int) (x!2 String)) Bool
    (let ((a!1 (not (or (BViaState x!0 x!1 x!2)
                        (BRecursive (+ (- 1) x!0) x!1 x!2))))))
      (not (or (not (>= x!0 1000)) a!1))))
  (define-fun B ((x!0 Int) (x!1 String)) Bool
    (ite (= (str.len x!1) 0) (= x!0 1000) (BRecursive 1008 x!0 x!1)))
  (define-fun A ((x!0 Int) (x!1 Int) (x!2 String)) Bool
    (ite (and (= x!0 1007) (= x!1 1005) (= x!2 "n")) true
      (ite (and (= x!0 1008) (= x!1 1005) (= x!2 "b")) true
        (ite (and (= x!0 1008) (= x!1 1007) (= x!2 "i")) true
          (ite (and (= x!0 1006) (= x!1 1001) (= x!2 "a")) true
            (ite (and (= x!0 1001) (= x!1 1005) (= x!2 "a")) true
              (ite (and (= x!0 1000) (= x!1 1001) (= x!2 "a")) true
                (ite (and (= x!0 1002) (= x!1 1008) (= x!2 "r")) true
                  (ite (and (= x!0 1008) (= x!1 1002) (= x!2 "o")) true
                    (ite (and (= x!0 1003) (= x!1 1008) (= x!2 "l")) true
                      (ite (and (= x!0 1000) (= x!1 1001) (= x!2 "f")) true
                        (ite (and (= x!0 1008) (= x!1 1006) (= x!2 "d")) true
                          (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "a")) true
                            (ite (and (= x!0 1000) (= x!1 1003) (= x!2 "f")) true
                              false))))))))))))))

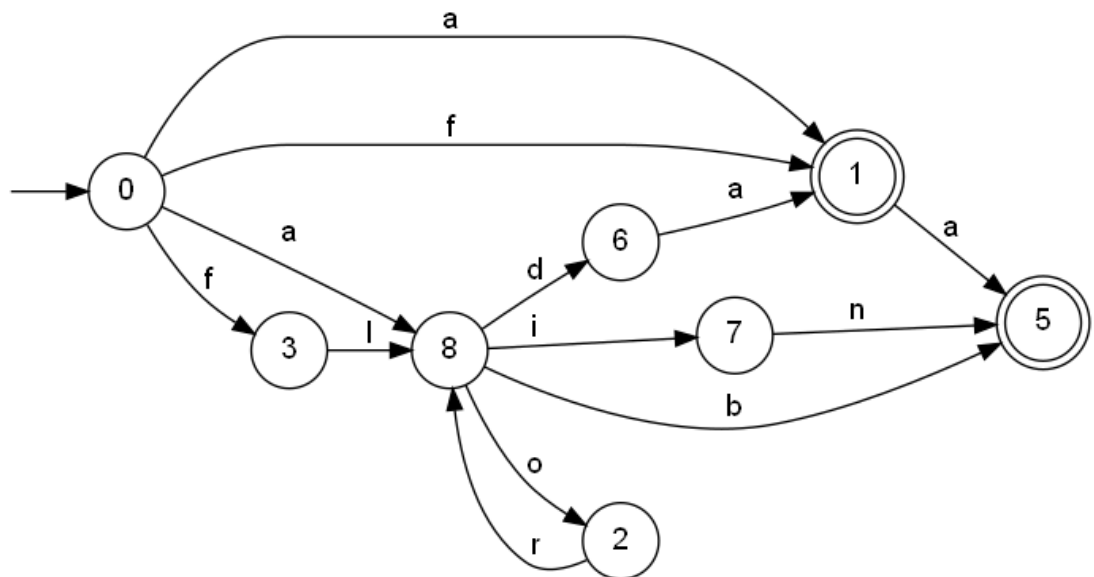
```

Parse tree:





Output:



Test 2

Input:

- *nfa.txt*:

```
(assert (and
  (Accept "ik")
  (Accept "hoi")
  (Accept "netherlands")
  (Accept "eindhoven")
  (Accept "hallo")
  (not (Accept ""))
  (not (Accept "hello"))
  (not (Accept "i"))
  (not (Accept "flor"))
  (not (Accept "bba"))
  (not (Accept "bbb")))
))

(check-sat)
(get-model)
```

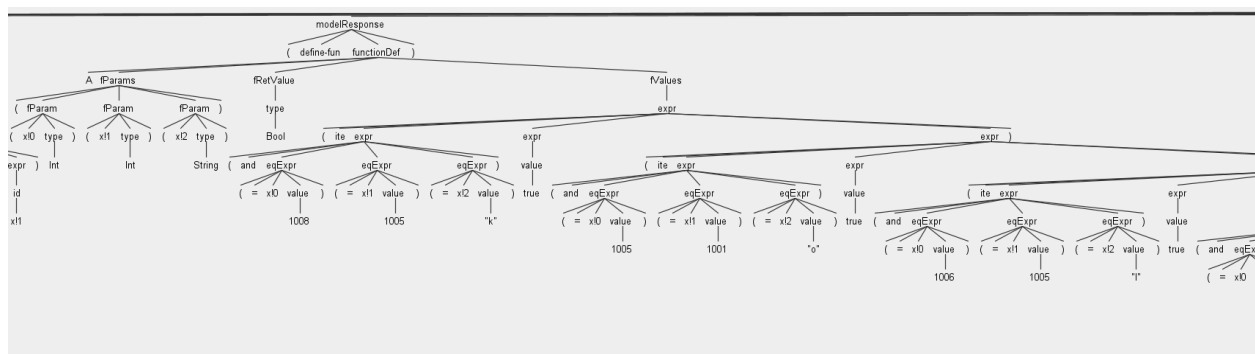
- *input.txt*:

```

"1001:1005:end")
(define-fun InitState () Int
  1000)
(define-fun MaxState () Int
  1008)
(define-fun BRecursive ((x!0 Int) (x!1 Int) (x!2 String)) Bool
  (let ((a!1 (not (or (BViaState x!0 x!1 x!2)
    (BRecursive (+ (- 1) x!0) x!1 x!2)))))
    (not (or (not (>= x!0 1000)) a!1))))
(define-fun B ((x!0 Int) (x!1 String)) Bool
  (ite (= (str.len x!1) 0) (= x!0 1000) (BRecursive 1008 x!0 x!1)))
(define-fun A ((x!0 Int) (x!1 Int) (x!2 String)) Bool
  (ite (and (= x!0 1008) (= x!1 1005) (= x!2 "k")) true
    (ite (and (= x!0 1005) (= x!1 1001) (= x!2 "o")) true
      (ite (and (= x!0 1006) (= x!1 1005) (= x!2 "l")) true
        (ite (and (= x!0 1008) (= x!1 1006) (= x!2 "l")) true
          (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "e")) true
            (ite (and (= x!0 1003) (= x!1 1005) (= x!2 "i")) true
              (ite (and (= x!0 1002) (= x!1 1003) (= x!2 "o")) true
                (ite (and (= x!0 1006) (= x!1 1007) (= x!2 "i")) true
                  (ite (and (= x!0 1008) (= x!1 1003) (= x!2 "h")) true
                    (ite (and (= x!0 1007) (= x!1 1008) (= x!2 "l")) true
                      (ite (and (= x!0 1005) (= x!1 1006) (= x!2 "o")) true
                        (ite (and (= x!0 1005) (= x!1 1006) (= x!2 "e")) true
                          (ite (and (= x!0 1006) (= x!1 1005) (= x!2 "h")) true
                            (ite (and (= x!0 1007) (= x!1 1008) (= x!2 "n")) true
                              (ite (and (= x!0 1002) (= x!1 1008) (= x!2 "a")) true
                                (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "i")) true
                                  (ite (and (= x!0 1007) (= x!1 1007) (= x!2 "n")) true
                                    (ite (and (= x!0 1003) (= x!1 1006) (= x!2 "e")) true
                                      (ite (and (= x!0 1000) (= x!1 1006) (= x!2 "e")) true
                                        (ite (and (= x!0 1000) (= x!1 1002) (= x!2 "h")) true
                                          (ite (and (= x!0 1006) (= x!1 1005) (= x!2 "n")) true
                                            (ite (and (= x!0 1008) (= x!1 1005) (= x!2 "s")) true
                                              (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "d")) true
                                                (ite (and (= x!0 1008) (= x!1 1007) (= x!2 "a")) true
                                                  (ite (and (= x!0 1006) (= x!1 1005) (= x!2 "v")) true
                                                    (ite (and (= x!0 1006) (= x!1 1007) (= x!2 "r")) true
                                                      (ite (and (= x!0 1007) (= x!1 1006) (= x!2 "d")) true
                                                        (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "t")) true
                                                          (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "n")) true
                                                            false))))))))))))))))))))))))))))))))))))))))))

```

Parse tree:



Test 3

Input:

- *nfa.txt*:

```
(assert (and  
  
  (Accept "fontys")  
  (Accept "eindhoven")  
  (Accept "campus")  
  (Accept "bike")  
  (Accept "hallo")  
  (not (Accept ""))  
  (not (Accept "tue"))  
  (not (Accept "automata"))  
  (not (Accept "flor"))  
  (not (Accept "bba"))  
  (not (Accept "bbb"))  
  
))
```

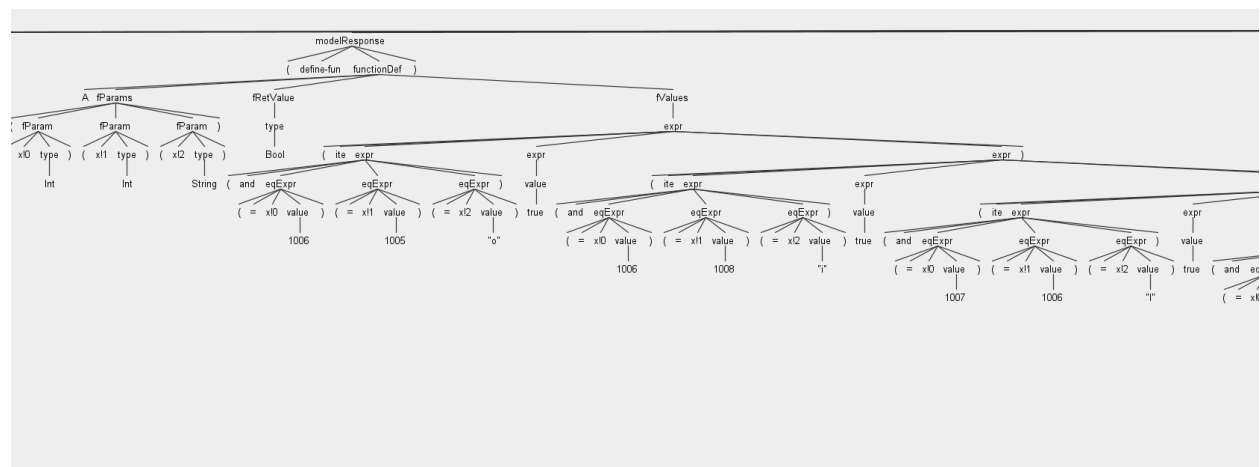
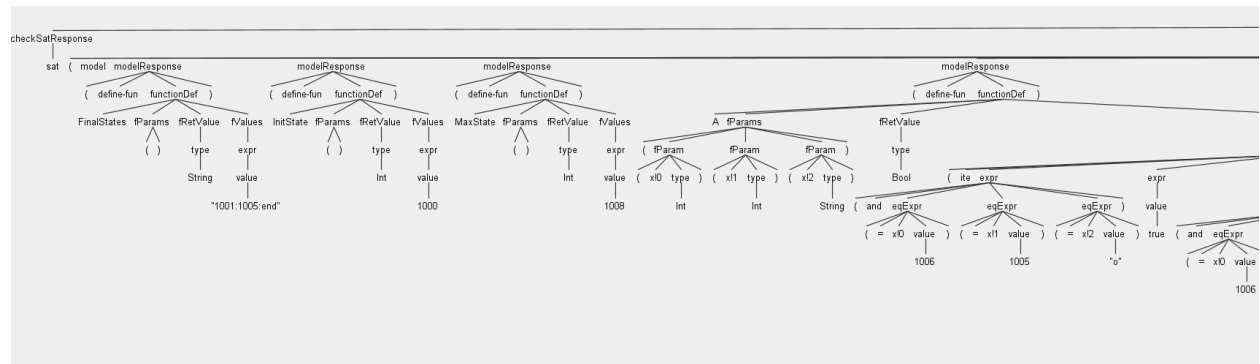
- *input.txt*:

```

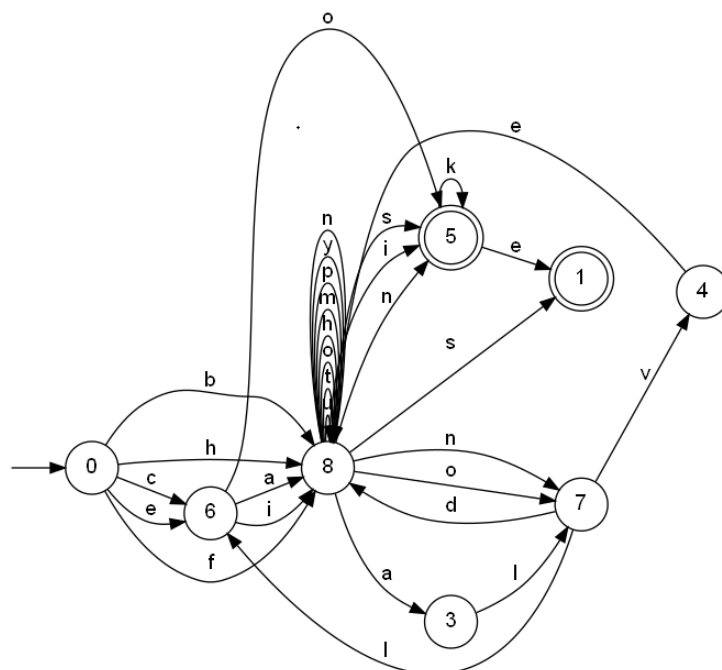
sat
(model
  (define-fun FinalStates () String
    "1001:1005:end")
  (define-fun InitState () Int
    1000)
  (define-fun MaxState () Int
    1008)
  (define-fun A ((x!0 Int) (x!1 Int) (x!2 String)) Bool
    (ite (and (= x!0 1006) (= x!1 1005) (= x!2 "o")) true
      (ite (and (= x!0 1006) (= x!1 1008) (= x!2 "i")) true
        (ite (and (= x!0 1007) (= x!1 1006) (= x!2 "l")) true
          (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "u")) true
            (ite (and (= x!0 1005) (= x!1 1001) (= x!2 "e")) true
              (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "t")) true
                (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "o")) true
                  (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "h")) true
                    (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "m")) true
                      (ite (and (= x!0 1003) (= x!1 1007) (= x!2 "l")) true
                        (ite (and (= x!0 1008) (= x!1 1003) (= x!2 "a")) true
                          (ite (and (= x!0 1008) (= x!1 1007) (= x!2 "o")) true
                            (ite (and (= x!0 1006) (= x!1 1008) (= x!2 "a")) true
                              (ite (and (= x!0 1008) (= x!1 1005) (= x!2 "i")) true
                                (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "b")) true
                                  (ite (and (= x!0 1008) (= x!1 1005) (= x!2 "n")) true
                                    (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "h")) true
                                      (ite (and (= x!0 1008) (= x!1 1005) (= x!2 "s")) true
                                        (ite (and (= x!0 1005) (= x!1 1005) (= x!2 "k")) true
                                          (ite (and (= x!0 1008) (= x!1 1001) (= x!2 "s")) true
                                            (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "p")) true
                                              (ite (and (= x!0 1000) (= x!1 1006) (= x!2 "c")) true
                                                (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "y")) true
                                                  (ite (and (= x!0 1008) (= x!1 1008) (= x!2 "n")) true
                                                    (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "f")) true
                                                      (ite (and (= x!0 1004) (= x!1 1008) (= x!2 "e")) true
                                                        (ite (and (= x!0 1007) (= x!1 1004) (= x!2 "v")) true
                                                          (ite (and (= x!0 1007) (= x!1 1008) (= x!2 "d")) true
                                                            (ite (and (= x!0 1008) (= x!1 1007) (= x!2 "n")) true
                                                              (ite (and (= x!0 1000) (= x!1 1006) (= x!2 "e")) true
                                                                false))))))))))))))))))))))))))))))

```

Parse tree:



Output:



Test 4

Input:

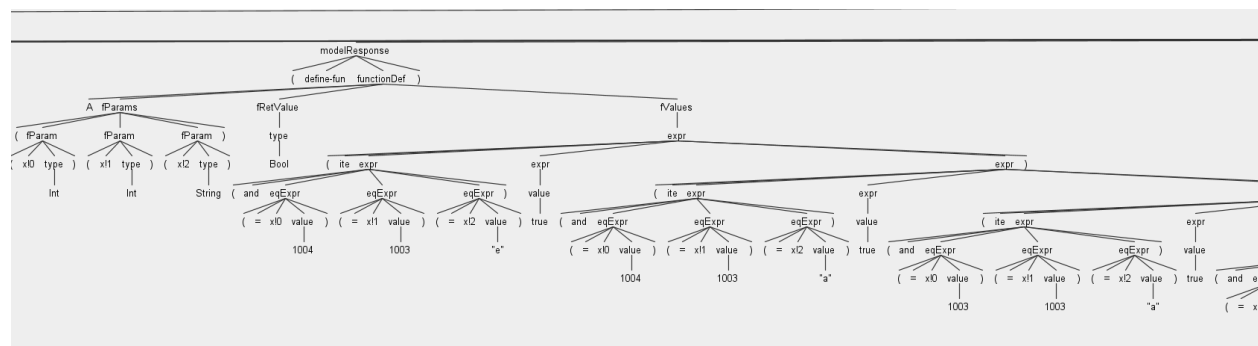
- *nfa.txt*:

```
(assert (and  
  
  (Accept "ada")  
  (Accept "deea")  
  (Accept "abe")  
  (Accept "dee")  
  (Accept "edeea")  
  (not (Accept ""))  
  (not (Accept "aae"))  
  (not (Accept "abb"))  
  (not (Accept "edd"))  
  (not (Accept "bba"))  
  (not (Accept "bbb"))  
  
))
```

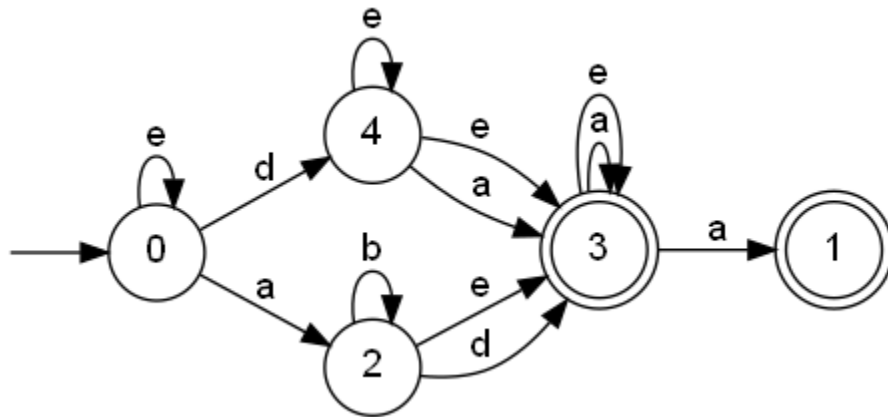
```
(declare-fun InitState () Int)  
(declare-fun MaxState () Int)  
(assert (= InitState 1000))  
(assert (= MaxState 1004))
```

- *input.txt*:

Parse tree:



Output:



Test 5

Input:

- *nfa.txt*:

```
(assert (and  
  
  (Accept "sare")  
  (Accept "tare")  
  (Accept "mare")  
  (Accept "mici")  
  (Accept "vama")  
  (Accept "mamaia")  
  (Accept "casino")  
  (not (Accept ""))  
  (not (Accept "piper"))  
  (not (Accept "mustar"))  
  (not (Accept "muraturi"))  
  (not (Accept "mic"))  
  
))
```

```

(declare-fun InitState () Int)
(declare-fun MaxState () Int)
(assert (= InitState 1000))
(assert (= MaxState 1011))

```

- *input.txt*:

```

sat
(model
  (define-fun FinalStates () String
    "1001:1002:1003:1007:1009:end")
  (define-fun InitState () Int
    1000)
  (define-fun MaxState () Int
    1011)
  (define-fun BRecursive ((x!0 Int) (x!1 Int) (x!2 String)) Bool
    (let ((a!1 (not (or (BViaState x!0 x!1 x!2)
                        (BRecursive (+ (- 1) x!0) x!1 x!2)))))
      (not (or (not (>= x!0 1000)) a!1))))
  (define-fun B ((x!0 Int) (x!1 String)) Bool
    (ite (= (str.len x!1) 0) (= x!0 1000) (BRecursive 1011 x!0 x!1)))
  (define-fun A ((x!0 Int) (x!1 Int) (x!2 String)) Bool
    (ite (and (= x!0 1011) (= x!1 1009) (= x!2 "a")) true
      (ite (and (= x!0 1010) (= x!1 1009) (= x!2 "e")) true
        (ite (and (= x!0 1011) (= x!1 1011) (= x!2 "i")) true
          (ite (and (= x!0 1011) (= x!1 1011) (= x!2 "a")) true
            (ite (and (= x!0 1008) (= x!1 1009) (= x!2 "i")) true
              (ite (and (= x!0 1010) (= x!1 1011) (= x!2 "m")) true
                (ite (and (= x!0 1008) (= x!1 1007) (= x!2 "i")) true
                  (ite (and (= x!0 1010) (= x!1 1010) (= x!2 "a")) true
                    (ite (and (= x!0 1011) (= x!1 1008) (= x!2 "c")) true
                      (ite (and (= x!0 1011) (= x!1 1010) (= x!2 "r")) true
                        (ite (and (= x!0 1010) (= x!1 1006) (= x!2 "a")) true
                          (ite (and (= x!0 1008) (= x!1 1011) (= x!2 "a")) true
                            (ite (and (= x!0 1007) (= x!1 1008) (= x!2 "a")) true
                              (ite (and (= x!0 1009) (= x!1 1008) (= x!2 "r")) true
                                (ite (and (= x!0 1000) (= x!1 1008) (= x!2 "t")) true
                                  (ite (and (= x!0 1008) (= x!1 1009) (= x!2 "e")) true
                                    (ite (and (= x!0 1006) (= x!1 1007) (= x!2 "m")) true
                                      (ite (and (= x!0 1004) (= x!1 1008) (= x!2 "r")) true
                                        (ite (and (= x!0 1011) (= x!1 1002) (= x!2 "s")) true
                                          (ite (and (= x!0 1004) (= x!1 1004) (= x!2 "a")) true
                                            (ite (and (= x!0 1008) (= x!1 1003) (= x!2 "e")) true
                                              (ite (and (= x!0 1002) (= x!1 1010) (= x!2 "i")) true
                                                (ite (and (= x!0 1009) (= x!1 1001) (= x!2 "a")) true
                                                  (ite (and (= x!0 1000) (= x!1 1011) (= x!2 "c")) true
                                                    (ite (and (= x!0 1000) (= x!1 1004) (= x!2 "s")) true
                                                      (ite (and (= x!0 1000) (= x!1 1010) (= x!2 "v")) true
                                                        (ite (and (= x!0 1000) (= x!1 1011) (= x!2 "m")) true

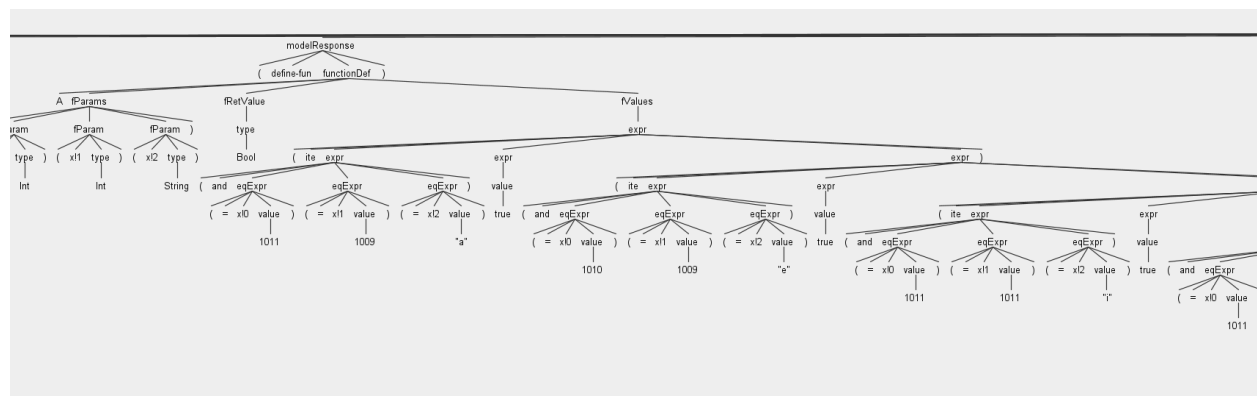
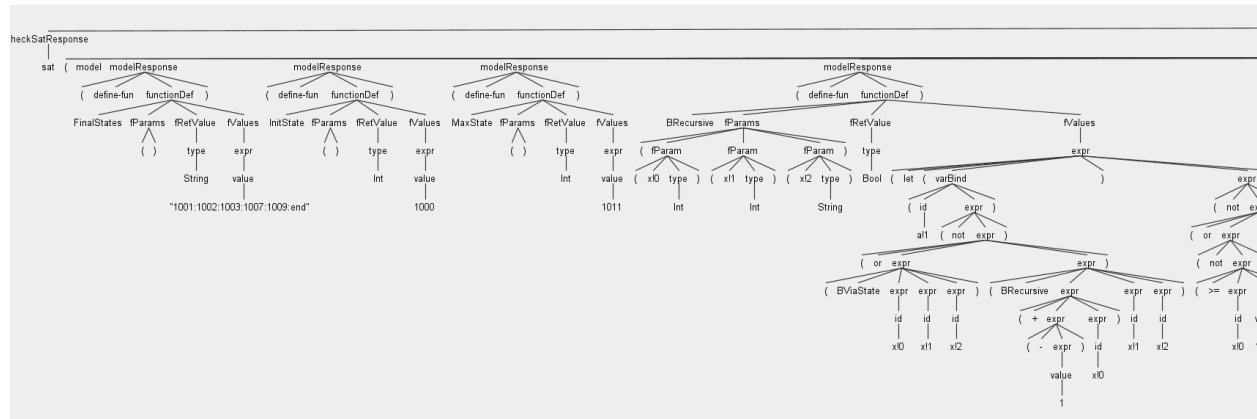
```

```

(ite (and (= x!0 1000) (= x!1 1011) (= x!2 "m"))) true
(ite (and (= x!0 1011) (= x!1 1009) (= x!2 "o"))) true
(ite (and (= x!0 1000) (= x!1 1010) (= x!2 "m"))) true
(ite (and (= x!0 1010) (= x!1 1011) (= x!2 "n"))) true
false))))))))))))))))))))))))))))))))))))))))))

```

Parse tree:



Output:

