

SSC0721 – Teste e inspeção de software

Teste Funcional – Exemplos de uso

Prof. Marcio E. Delamaro

`delamaro@icmc.usp.br`

Relembrando

- Foram vistos três critérios de teste funcional
- Serão vistos três exemplos de como podem ser aplicados esses critérios
- retirados do livro do Mayers
- Exercício
- Trabalho

Especificação 1

- Queremos construir um compilador FORTRAN. Vamos validar a análise de comandos **DIMENSION** (arrays).
- A forma do comando é:
`DIMENSION ad[,ad]...`
- *ad* é um descritor de array da forma
`n(d[,d]...)`
- *n* é o nome do array e *d* é a dimensão
- O nome pode ter de um a seis letras ou dígitos, sendo o primeiro caractere uma letra, obrigatoriamente

Especificação 1

- Cada array pode ter de um a sete dimensões
- Cada declaração de dimensão tem a forma
[*lb*: *ub*]
- *lb* é o limite inferior e *ub* o superior
- Limites variam entre -65534 e 65535 ou podem ser o nome de uma variável inteira (mas não o elemento de um array).
- Se *lb* não for especificado seu valor é 1
- *ub* tem que ser maior ou igual a *lb*
- *lb* pode ser negativo, positivo ou zero
- Como qualquer comando, a declaração DIMENSION pode tomar diversas linhas do programa

Especificação 1

```
DIMENSION A(2)
```

```
DIMENSION A12345 (I,9, J4XXXX,65535,1,KLM,  
X100), BBB(-65534:100,0:1000,10:10, I:65535)
```

Definição das condições de entrada

Condição de entrada	Classes válidas	Classes inválidas

Definição das condições de entrada

Condição de entrada	Classes válidas	Classes inválidas
# de descritores		
Tamanho do nome		
Nome do array		
Nome começa letra		
# de dimensões		
Limite sup é		
Nome de var tem		
Var começa letra		

Definição das condições de entrada

Condição de entrada	Classes válidas	Classes inválidas
Constante		
lb especificado		
up \times lb		
valor de lb		
lb é		
multiplas linha		

Definição das condições de entrada

Condição de entrada	Classes válidas	Classes inválidas
# de descritores	um (1) > um (2)	zero (3)
Tamanho do nome	1–6 (4)	zero(5) > 6 (6)
Nome do array	tem letra (7) tem dígito (8)	outra coisa (9)
Nome começa letra	sim (10)	não (11)
# de dimensões	1–7 (12)	zero (13) >7 (14)
Limite sup é	const(15) var (16)	array elem (17) outra coisa (18)
Nome de var	tem letra (19) tem dígito (20)	outra coisa (21)
Var começa letra	sim (22)	não (23)

Definição das condições de entrada

Condição de entrada	Classes válidas	Classes inválidas
Constante	-65534 a 65535 (24)	< 65534 (25) > 65535 (26)
lb especificado	S (27) N (28)	
up × lb	maior (29) menor (30)	< menor (31)
valor de lb	< 0 (32) 0 (33) > 0 (34)	
lb é	const (35) var (36)	array elem (37) outra coisa (38)
multiplas linha	S (39) N (40)	

Casos de teste – válidas

- DIMENSION A(2)

- 1, 4, 7, 10, 12, 15, 24, 28, 29, 40

- DIMENSION A(2)

DIMENSION A12345 (I,9, J4XXXX,65535,1,KLM,
X100), BBB(-65534:100,0:1000,10:10,
I:65535)

- 2, 8, 16, 19, 20, 22, 27, 30, 32, 33, 34, 35, 36, 39

Casos de teste – não válidos

- (3): DIMENSION
- (5): DIMENSION (10)
- (6): DIMENSION A234567(2)
- (9): DIMENSION A.1(2)
- (11): DIMENSION 1A(10)
- (13): DIMENSION B
- (14): DIMENSION B(4,4,4,4,4,4,4,4)
- (17): DIMENSION B(4,A(2))

Casos de teste – não válidos

- (18): DIMENSION B(4,,7)
- (21): DIMENSION C(I.,10)
- (23): DIMENSION C(10,1J)
- (25): DIMENSION D(-65535:1)
- (26): DIMENSION D(65536)
- (31): DIMENSION D(4:3)
- (37): DIMENSION D(A(2):4)
- (38): D(.:4)

Especificação 2

- MTEST é um programa que dá notas em testes de múltipla escolha
- A entrada é um arquivo texto chamado OCR com diversos registros de tamanho 80 (caracteres)
- Em cada arquivo, o primeiro registro contém um título, usado nos relatórios produzidos pelo programa

Especificação 2

- O segundo conjunto de registros descreve as respostas corretas do teste.
- Todos registros possuem “2” como último caractere (coluna 80)
- No primeiro registro, colunas 1 a 3, está o número de questões (1–999)
- Colunas 10 a 59 contêm respostas para as questões 1 a 50
- Qualquer caractere é válido como resposta.
- Registros subsequentes possuem as respostas para as questões 51 a 100, 101 a 150, etc, sempre nas colunas 10 a 59

Especificação 2

- O terceiro conjunto de registros contém as respostas dos alunos
- Cada um tem o caractere "3" na coluna 80
- Para cada estudante, o primeiro registro tem nome ou número nas posições 1 a 9 (qualquer caractere)
- As posições 10 a 59 contém as respostas do estudante para as questões 1 a 50
- Se houver mais questões, elas estão nos registros seguintes, sempre nas colunas 10 a 59
- O número máximo de estudantes é 200

Especificação 2 – registros de entrada

Title										80
No. of questions			Correct answers 1–50					2		
1	3	4	9	10	59	60	79	80		
			Correct answers 51–100					2		
1			9	10	59	60	79	80		
Student identifier			Correct answers 1–50					3		
1			9	10	59	60	79	80		
			Correct answers 51–100					3		
1			9	10	59	60	79	80		

Especificação 2 – saídas

- Um relatório, ordenado pelo identificar do estudante, mostrando a nota (porcentagem de acerto) e a classificação
- Relatório similar mas ordenado por nota
- Relatório indicando a média, a mediana e o desvio padrão das notas
- Um relatório, ordenado pelo número das questões mostrando a porcentagem de estudantes que acertaram cada uma

Condições e limites (1)

- Tamanho do arquivo: vazio

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente
 - Maior e menor títulos possíveis

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente
 - Maior e menor títulos possíveis
- Presença do registro de respostas e o número de questões

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente
 - Maior e menor títulos possíveis
- Presença do registro de respostas e o número de questões
 - Classes de equivalência: 1-50 e 51-999

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente
 - Maior e menor títulos possíveis
- Presença do registro de respostas e o número de questões
 - Classes de equivalência: 1-50 e 51-999
 - Ausência de registros de resposta

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente
 - Maior e menor títulos possíveis
- Presença do registro de respostas e o número de questões
 - Classes de equivalência: 1-50 e 51-999
 - Ausência de registros de resposta
 - Um registro a mais do que o especificado

Condições e limites (1)

- Tamanho do arquivo: vazio
- Registro de título
 - Ausente
 - Maior e menor títulos possíveis
- Presença do registro de respostas e o número de questões
 - Classes de equivalência: 1-50 e 51-999
 - Ausência de registros de resposta
 - Um registro a mais do que o especificado
 - Um registro a menos

Casos de teste (1)

- 1. Arquivo de entrada vazio
- 2. Registro de título ausente
- 3. Título com 1 caractere
- 4. Título com 80 caracteres
- 5. Prova com apenas 1 questão
- 6. Prova com 50 questões
- 7. Prova com 51 questões

Casos de teste (1)

- 8. Prova com 999 questões
- 9. Prova com nenhuma questão
- 10. Campo com número de questões tem não-número
- 11. Nenhum registro de respostas após o título
- 12. Um registro a mais de respostas
- 13. Um registro a menos de respostas

Condições e limites e casos de teste(2)

- Registros de respostas

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes
 - 18. Estudante tem 1 registro de resposta mas existem 2 registros de respostas corretas

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes
 - 18. Estudante tem 1 registro de resposta mas existem 2 registros de respostas corretas
 - 19. Esse estudante é o primeiro

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes
 - 18. Estudante tem 1 registro de resposta mas existem 2 registros de respostas corretas
 - 19. Esse estudante é o primeiro
 - 20. Esse estudante é o último

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes
 - 18. Estudante tem 1 registro de resposta mas existem 2 registros de respostas corretas
 - 19. Esse estudante é o primeiro
 - 20. Esse estudante é o último
 - 21. Estudante tem 2 registros de resposta mas existe 1 registro de respostas corretas

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes
 - 18. Estudante tem 1 registro de resposta mas existem 2 registros de respostas corretas
 - 19. Esse estudante é o primeiro
 - 20. Esse estudante é o último
 - 21. Estudante tem 2 registros de resposta mas existe 1 registro de respostas corretas
 - 22. Esse estudante é o primeiro

Condições e limites e casos de teste(2)

- Registros de respostas
 - 14. Nenhum estudante
 - 15. 1 estudante
 - 16. 200 estudantes
 - 17. 201 estudantes
 - 18. Estudante tem 1 registro de resposta mas existem 2 registros de respostas corretas
 - 19. Esse estudante é o primeiro
 - 20. Esse estudante é o último
 - 21. Estudante tem 2 registros de resposta mas existe 1 registro de respostas corretas
 - 22. Esse estudante é o primeiro
 - 23. Esse estudante é o último

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - nenhum, 1, 200 estudantes (já existem)

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - nenhum, 1, 200 estudantes (já existem)
 - 24. Todos estudantes recebem mesma nota

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - nenhum, 1, 200 estudantes (já existem)
 - 24. Todos estudantes recebem mesma nota
 - 25. Todos estudantes recebem notas diferentes

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - nenhum, 1, 200 estudantes (já existem)
 - 24. Todos estudantes recebem mesma nota
 - 25. Todos estudantes recebem notas diferentes
 - 26. Alguns, mas não todos, recebem mesma nota (verificar se classificação é computada certo)

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - nenhum, 1, 200 estudantes (já existem)
 - 24. Todos estudantes recebem mesma nota
 - 25. Todos estudantes recebem notas diferentes
 - 26. Alguns, mas não todos, recebem mesma nota (verificar se classificação é computada certo)
 - 27. Um estudante recebe 0

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - nenhum, 1, 200 estudantes (já existem)
 - 24. Todos estudantes recebem mesma nota
 - 25. Todos estudantes recebem notas diferentes
 - 26. Alguns, mas não todos, recebem mesma nota (verificar se classificação é computada certo)
 - 27. Um estudante recebe 0
 - 28. Um estudante recebe 100

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - 29. Um estudante tem o menor identificador possível (verificar a ordenação)

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - 29. Um estudante tem o menor identificador possível (verificar a ordenação)
 - 30. Um estudante tem o maior identificador possível

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - 29. Um estudante tem o menor identificador possível (verificar a ordenação)
 - 30. Um estudante tem o maior identificador possível
 - 31. O número de estudantes faz com que o relatório caiba exatamente em uma página

Condições de saída e casos de teste

- Relatórios 1 e 2 (alunos \times notas)
 - 29. Um estudante tem o menor identificador possível (verificar a ordenação)
 - 30. Um estudante tem o maior identificador possível
 - 31. O número de estudantes faz com que o relatório caiba exatamente em uma página
 - 32. O número de estudantes faz com que todos, exceto um, caibam em uma única página

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)
 - 33. Maior valor da média (todos com 100)

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)
 - 33. Maior valor da média (todos com 100)
 - 34. A média é zero (todos com 0)

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)
 - 33. Maior valor da média (todos com 100)
 - 34. A média é zero (todos com 0)
 - 35. Desvio padrão é o máximo (um 0 e um 100)

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)
 - 33. Maior valor da média (todos com 100)
 - 34. A média é zero (todos com 0)
 - 35. Desvio padrão é o máximo (um 0 e um 100)
 - 36. Desvio padrão é 0 (todos com mesma nota – já existe???)

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)
 - 33. Maior valor da média (todos com 100)
 - 34. A média é zero (todos com 0)
 - 35. Desvio padrão é o máximo (um 0 e um 100)
 - 36. Desvio padrão é 0 (todos com mesma nota – já existe???)
- 33 e 34 cobrem limites da mediana

Condições de saída e casos de teste

- Relatório 3 (média, mediana, desvio padrão)
 - 33. Maior valor da média (todos com 100)
 - 34. A média é zero (todos com 0)
 - 35. Desvio padrão é o máximo (um 0 e um 100)
 - 36. Desvio padrão é 0 (todos com mesma nota – já existe???)
- 33 e 34 cobrem limites da mediana
- Nenhum estudante para tentar divisão por zero!!! (já existe)

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)
 - 37. Todos acertam questão 1

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)
 - 37. Todos acertam questão 1
 - 38. Todos erram questão 1

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)
 - 37. Todos acertam questão 1
 - 38. Todos erram questão 1
 - 39. Todos acertam última questão

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)
 - 37. Todos acertam questão 1
 - 38. Todos erram questão 1
 - 39. Todos acertam última questão
 - 40. Todos erram última questão

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)
 - 37. Todos acertam questão 1
 - 38. Todos erram questão 1
 - 39. Todos acertam última questão
 - 40. Todos erram última questão
 - 41. Número de questões faz com que relatório caiba exatamente numa página

Condições de saída e casos de teste

- Relatório 4 (acertos por questão)
 - 37. Todos acertam questão 1
 - 38. Todos erram questão 1
 - 39. Todos acertam última questão
 - 40. Todos erram última questão
 - 41. Número de questões faz com que relatório caiba exatamente numa página
 - 42. Número de questões faz com que todas as questões, exceto uma, caibam em uma página

Especificação 3

- Depurador possui um comando "DISPLAY"
- Argumentos são os seguintes:

$$DISPLAY \left[\begin{array}{c} hexloc \\ \underline{0} \end{array} \right] \left[\begin{array}{c} -hexloc \\ -END \\ .bytecount \\ \underline{1} \end{array} \right]$$

- Maiúsculo: palavra chave
- Minúsculas: valores a serem fornecidos
- Sublinhado: valor default do parâmetro
- Colchetes: opcional

Argumentos

- *hexloc*: um a seis dígitos hexa. Indica a posição inicial de memória a ser exibida.
- Se não for fornecida, o valor é 0
- Precisa estar dentro dos limites de memória do programa sendo tratado
- O segundo operando especifica a quantidade de memória a ser mostrada
- *hexloc2*: um a seis dígitos hexa. Indica o endereço do último byte a ser exibido
- Deve ser maior ou igual a *hexloc*
- Precisa estar dentro dos limites de memória do programa sendo tratado

Argumentos

- -END indica até o fim do program
- *bytecountum* a seis dígitos hexa. Indica o número de bytes a serem exibidos
- Sua soma com o endereço inicial não deve exceder o tamanho da memória + 1
- Deve ser maior ou igual a 1

- Uma ou mais linhas
`xxxxxx = word1 word2 word3 word4`
- Uma palavra é uma sequência de 4 bytes. O endereço do primeiro byte é sempre múltiplo de 4
- O primeiro byte do intervalo definido pelos argumentos deve estar na primeira palavra da linha
- Todas as linhas mostram quatro palavras

- Mensagens de erro
 - M1: sintaxe do comando inválida
 - M2: memória requisitada está além dos limites da memória existente
 - M3: Memória requisitada é um intervalo nulo ou negativo

- DISPLAY
000000 = 023f6790 002897ca cafebabe 000f0b10
- DISPLAY 77F
00077c = 67899ff1 00000000 00000000 00000000
- DISPLAY 77F.6
00077c = 67899ff1 783fa4fa 8aa9b133 00000000

- 1. Primeiro operando está presente

Causas

- 1. Primeiro operando está presente
- 2. hexloc contém apenas dígitos hexadecimais

Causas

- 1. Primeiro operando está presente
- 2. hexloc contém apenas dígitos hexadecimais
- 3. hexloc contém entre 1 e 6 caracteres

Causas

- 1. Primeiro operando está presente
- 2. hexloc contém apenas dígitos hexadecimais
- 3. hexloc contém entre 1 e 6 caracteres
- 4. hexloc está dentro do limite de memória existente

Causas

- 1. Primeiro operando está presente
- 2. hexloc contém apenas dígitos hexadecimais
- 3. hexloc contém entre 1 e 6 caracteres
- 4. hexloc está dentro do limite de memória existente
- 5. Segundo argumento é END

Causas

- 1. Primeiro operando está presente
- 2. hexloc contém apenas dígitos hexadecimais
- 3. hexloc contém entre 1 e 6 caracteres
- 4. hexloc está dentro do limite de memória existente
- 5. Segundo argumento é END
- 6. Segundo argumento é hexloc2

Causas

- 1. Primeiro operando está presente
- 2. hexloc contém apenas dígitos hexadecimais
- 3. hexloc contém entre 1 e 6 caracteres
- 4. hexloc está dentro do limite de memória existente
- 5. Segundo argumento é END
- 6. Segundo argumento é hexloc2
- 7. Segundo operando é bytecount

- 8. Segundo operando não presente

- 8. Segundo operando não presente
- 9. hexloc2 contém apenas caracteres hexadecimais

- 8. Segundo operando não presente
- 9. hexloc2 contém apenas caracteres hexadecimais
- 10. hexlo2c contém entre 1 e 6 caracteres

- 8. Segundo operando não presente
- 9. hexloc2 contém apenas caracteres hexadecimais
- 10. hexlo2c contém entre 1 e 6 caracteres
- 11. hexloc2 está dentro do limite de memória existente

Causas

- 8. Segundo operando não presente
- 9. hexloc2 contém apenas caracteres hexadecimais
- 10. hexlo2c contém entre 1 e 6 caracteres
- 11. hexloc2 está dentro do limite de memória existente
- 12. $\text{hexloc2} \geq \text{hexloc}$

- 8. Segundo operando não presente
- 9. hexloc2 contém apenas caracteres hexadecimais
- 10. hexlo2c contém entre 1 e 6 caracteres
- 11. hexloc2 está dentro do limite de memória existente
- 12. $\text{hexloc2} \geq \text{hexloc}$
- 13. bytecount contém apenas caracteres hexadecimais

Causas

- 8. Segundo operando não presente
- 9. hexloc2 contém apenas caracteres hexadecimais
- 10. hexlo2c contém entre 1 e 6 caracteres
- 11. hexloc2 está dentro do limite de memória existente
- 12. $\text{hexloc2} \geq \text{hexloc}$
- 13. bytecount contém apenas caracteres hexadecimais
- 14. bytecount contém entre 1 e 6 caracteres

- 15. $\text{bytecount} + \text{hexloc1} \leq \text{memory size} + 1$.

- 15. $\text{bytecount} + \text{hexloc1} \leq \text{memory size} + 1$.
- 16. $\text{bytecount} \geq 1$.

- 15. $\text{bytecount} + \text{hexloc1} \leq \text{memory size} + 1$.
- 16. $\text{bytecount} \geq 1$.
- 17. Intervalo especificado requer múltiplas linhas de saída

Causas

- 15. $\text{bytecount} + \text{hexloc1} \leq \text{memory size} + 1$.
- 16. $\text{bytecount} \geq 1$.
- 17. Intervalo especificado requer múltiplas linhas de saída
- 18. Início do intervalo não cai no início de uma palavra

- 91. Mensagem M1 é apresentada

- 91. Mensagem M1 é apresentada
- 92. Mensagem M2 é apresentada

- 91. Mensagem M1 é apresentada
- 92. Mensagem M2 é apresentada
- 93. Mensagem M3 é apresentada

Efeitos

- 91. Mensagem M1 é apresentada
- 92. Mensagem M2 é apresentada
- 93. Mensagem M3 é apresentada
- 94. Memória é apresentada em uma única linha

- 91. Mensagem M1 é apresentada
- 92. Mensagem M2 é apresentada
- 93. Mensagem M3 é apresentada
- 94. Memória é apresentada em uma única linha
- 95. Memória é apresentada em múltiplas linhas

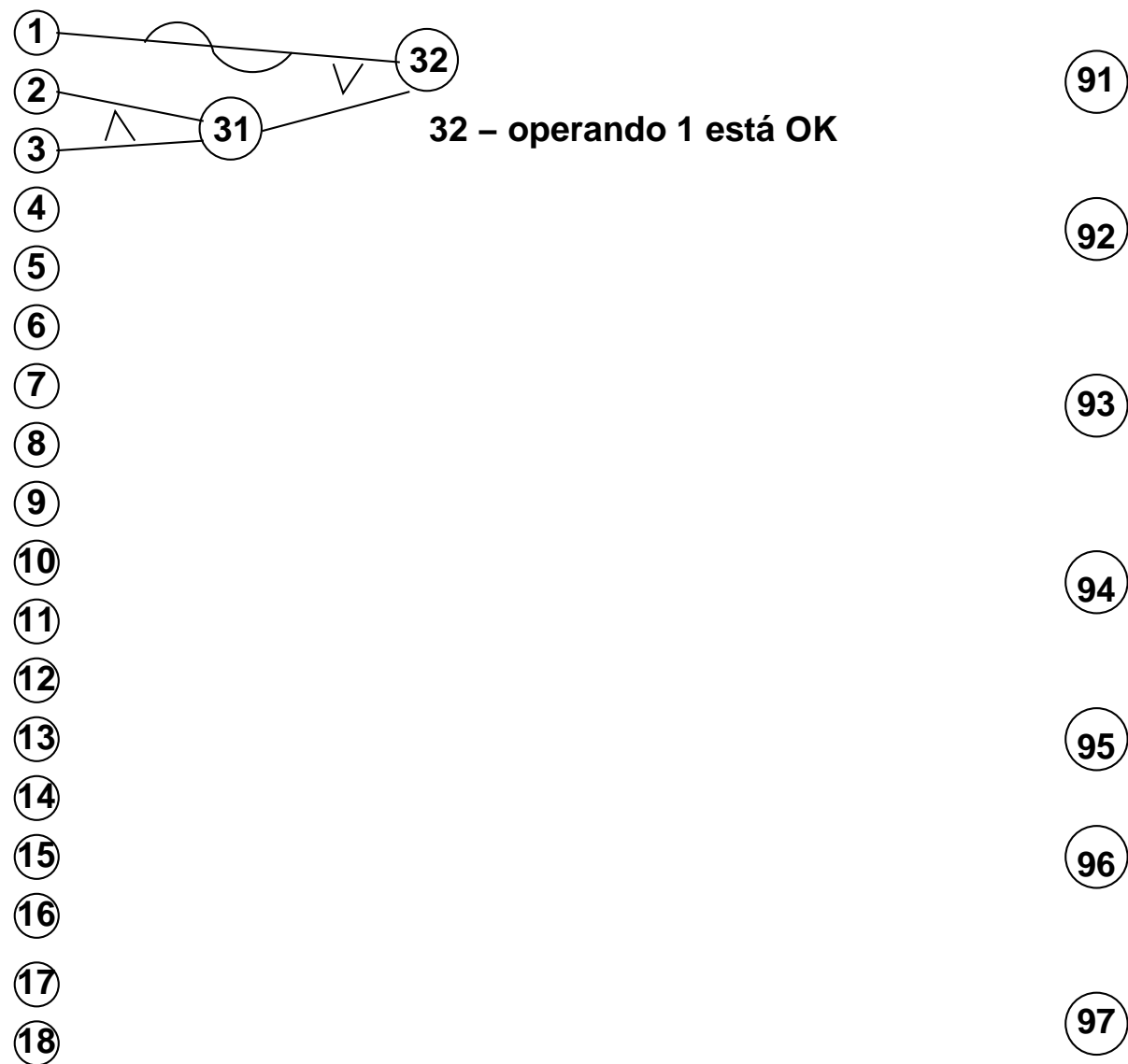
Efeitos

- 91. Mensagem M1 é apresentada
- 92. Mensagem M2 é apresentada
- 93. Mensagem M3 é apresentada
- 94. Memória é apresentada em uma única linha
- 95. Memória é apresentada em múltiplas linhas
- 96. Primeiro byte do intervalo cai no início de uma palavra

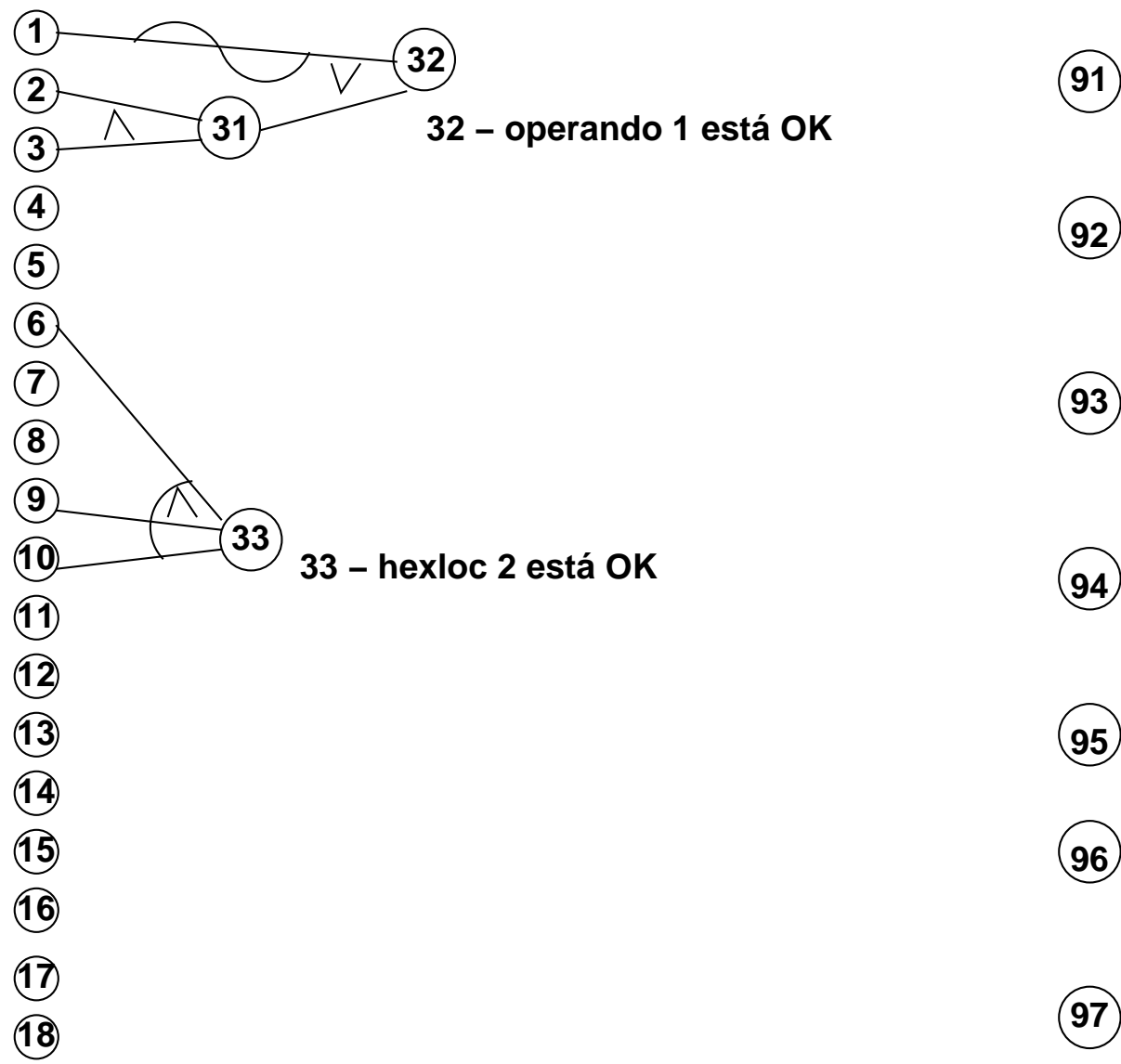
Efeitos

- 91. Mensagem M1 é apresentada
- 92. Mensagem M2 é apresentada
- 93. Mensagem M3 é apresentada
- 94. Memória é apresentada em uma única linha
- 95. Memória é apresentada em múltiplas linhas
- 96. Primeiro byte do intervalo cai no início de uma palavra
- 97. Primeiro byte do intervalo não cai no início de uma palavra

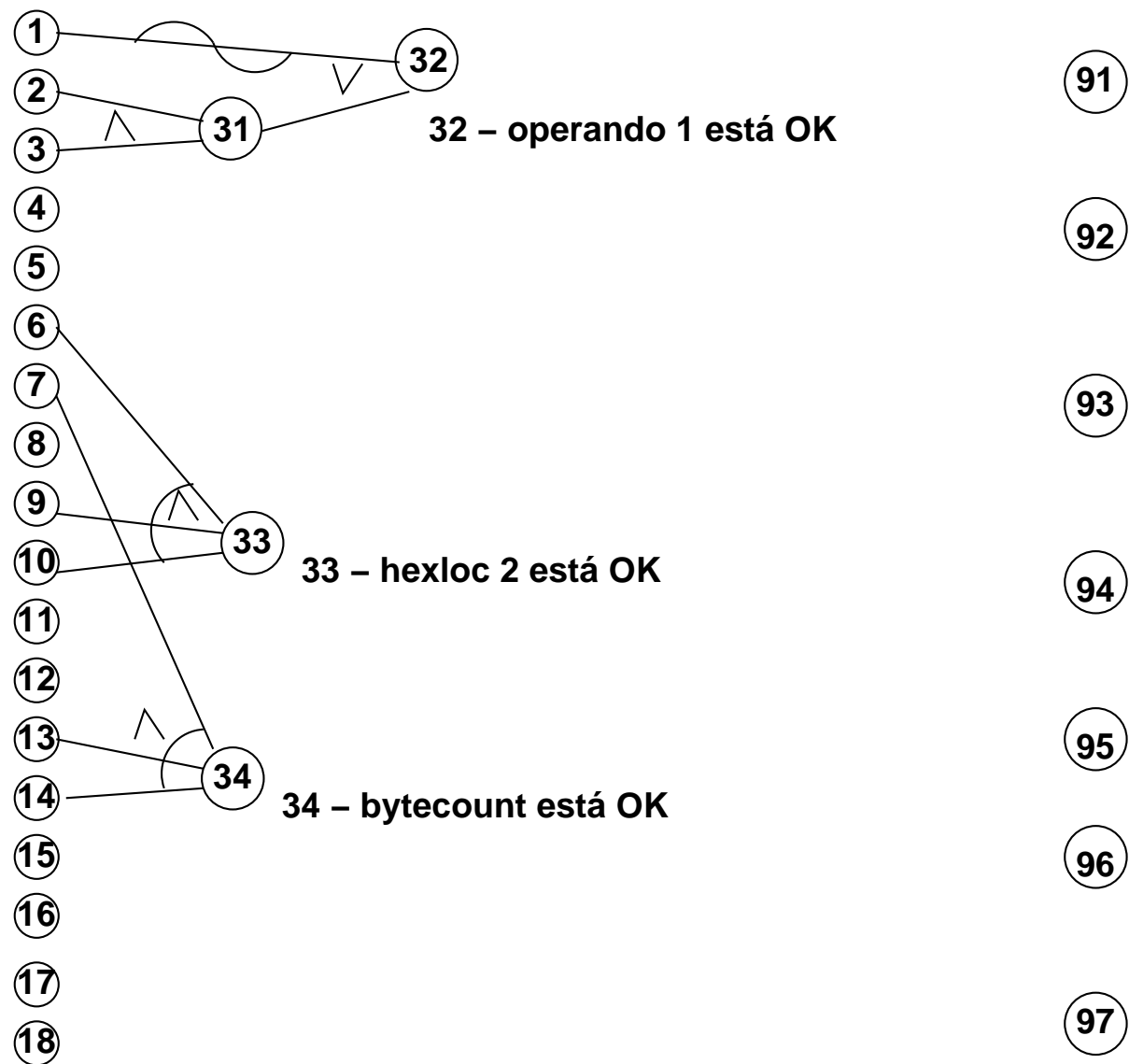
Construção do grafo



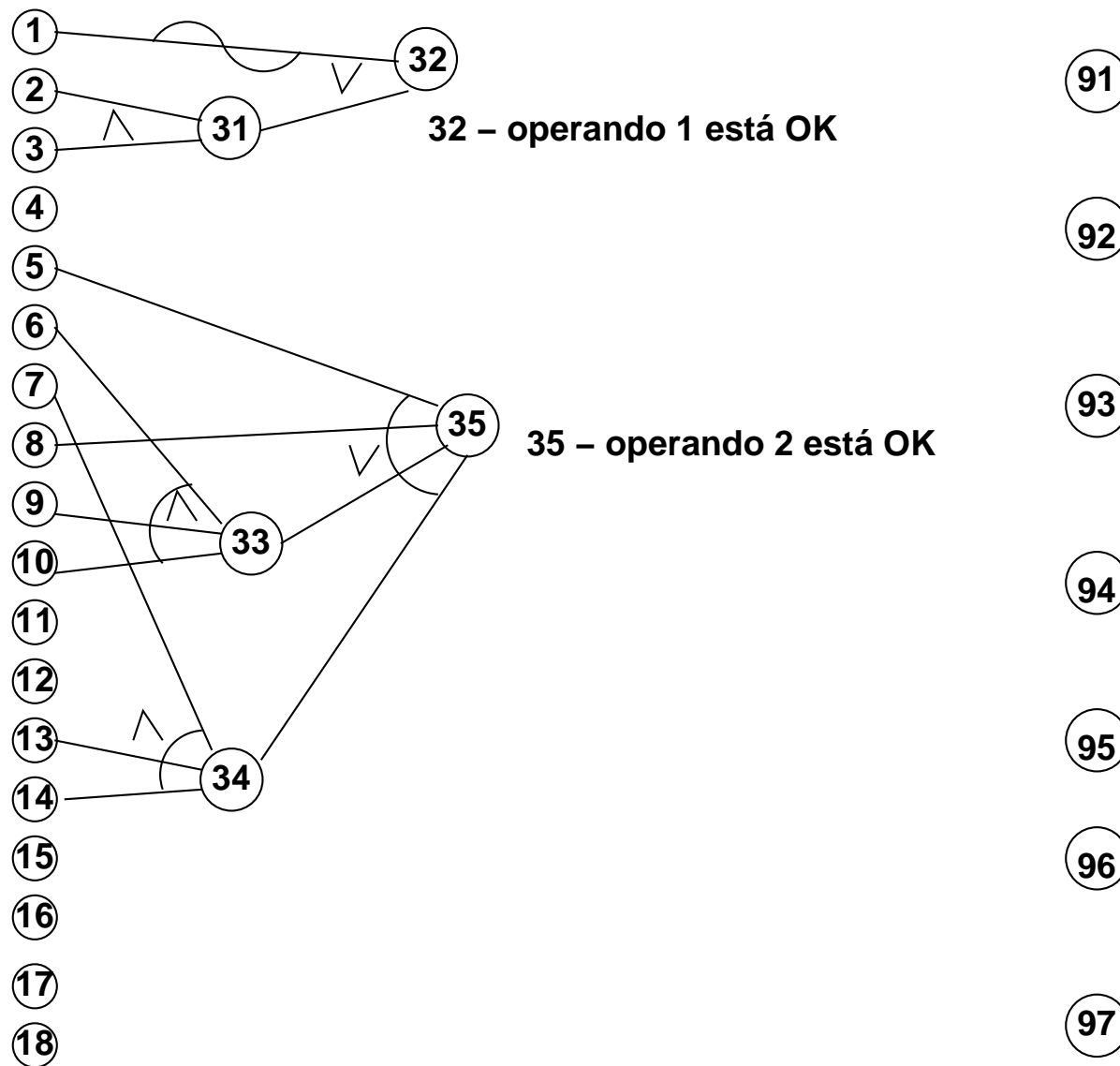
Construção do grafo



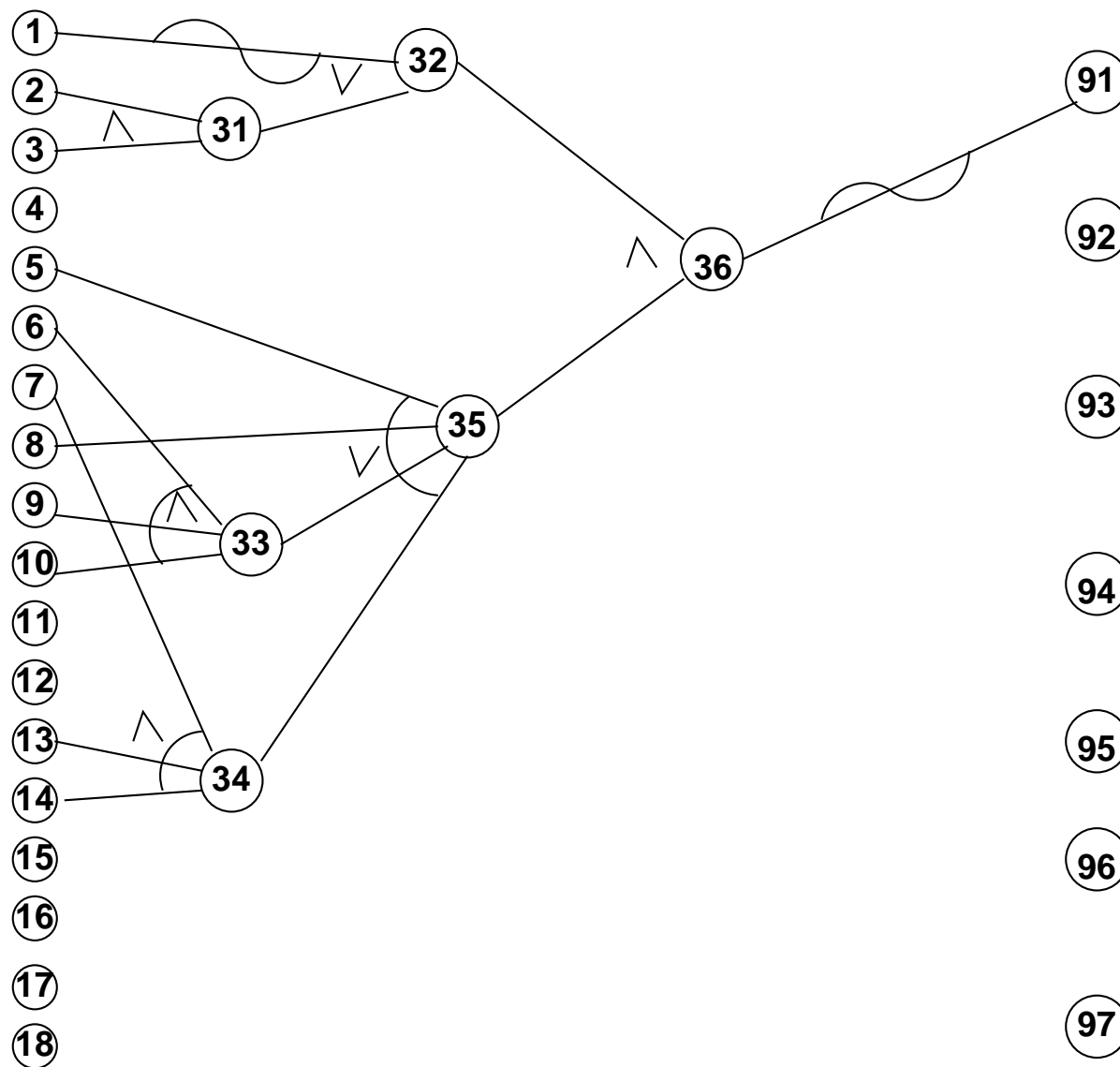
Construção do grafo



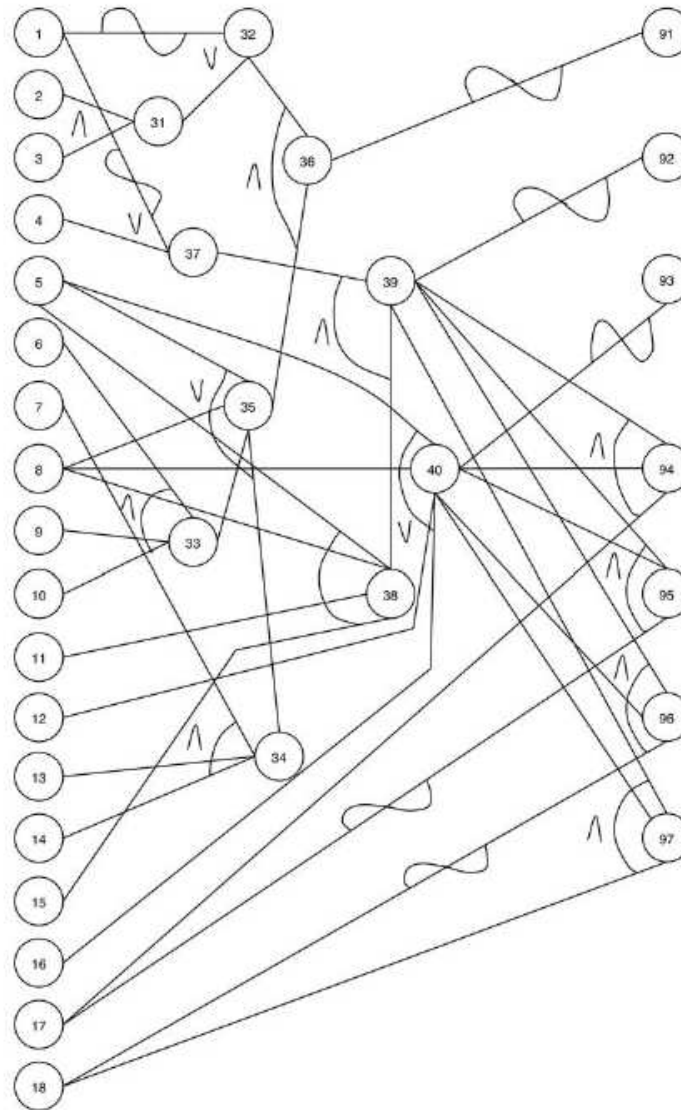
Construção do grafo



Construção do grafo



Construção do grafo



Restrições

- Evitar que casos de teste impossíveis sejam gerados
- Causas 2 e 3 não podem estar presentes a não ser que a causa 1 também esteja
- Causa 4 não pode estar presente a não ser que causas 2 e 3 estejam ambas presentes
- No máximo uma das causas 5, 6 e 7 pode estar presente
- Muitas restrições “requer”.
- Causa 17 (múltiplas linhas) requer a negação da causa 8 (segundo operando é omitido)

Grafo com restrições

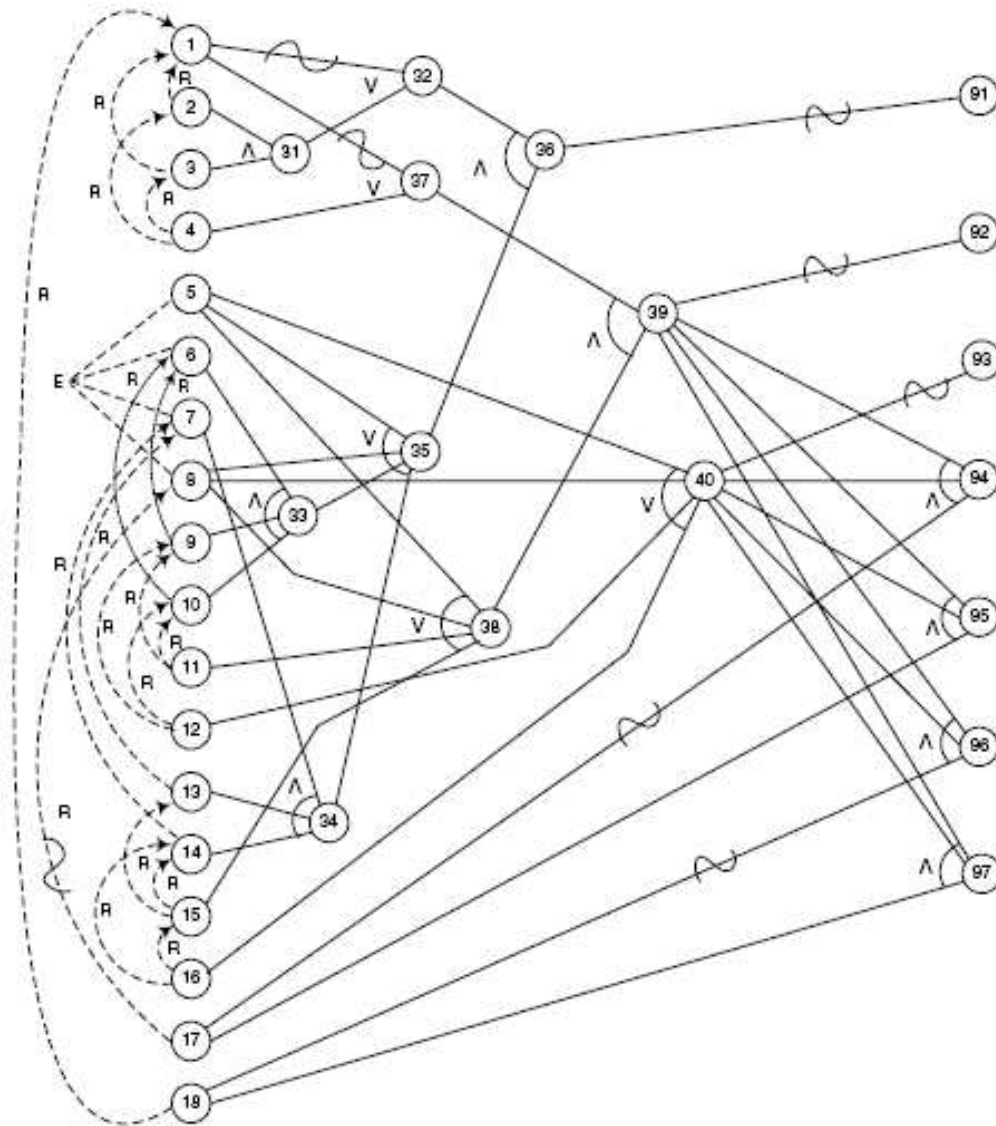


Tabela de decisão

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1
4												1	1	0	0	1	1
5				0										1			
6	1	1	1	0	1	1	1				1	1			1	1	
7				0				1	1	1			1				1
8				0													
9	1	1	1		1	0	0				0	1			1	1	
10	1	1	1		0	1	0				1	1			1	1	
11												0			0	1	
12																0	
13								1	0	0			1				1
14								0	1	0			1				1
15													0				
16																	0
17																	
18																	
91	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
92	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0
93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
94	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
96	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabela de decisão

	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
1	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	0	0	0	1	1	1
2	1	1	1	1					1	1	1	1	1	1	1				1	1	1
3	1	1	1	1					1	1	1	1	1	1	1				1	1	1
4	1	1	1	1					1	1	1	1	1	1	1				1	1	1
5	1				1				1				1			1			1		
6			1				1				1			1			1			1	
7				1				1				1			1			1			1
8		1				1				1											
9			1				1				1			1			1			1	
10			1				1				1			1			1			1	
11			1				1				1			1			1			1	
12			1				1				1			1			1			1	
13				1				1				1			1			1			1
14				1				1				1			1			1			1
15				1				1				1			1			1			1
16				1				1				1			1			1			1
17	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
18	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
92	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
94	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
95	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
96	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1
97	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0

Casos de teste

- Último endereço válido: 7fff
- 1 DISPLAY 234AF74-123 (91)
- 2 DISPLAY 2ZX4-3000 (91)
- 3 DISPLAY HHHHHHHH-2000 (91)
- 4 DISPLAY 200 200 (91)
- 5 DISPLAY 0-22222222 (91)
- 6 DISPLAY 1-2X (91)
- 7 DISPLAY 2-ABCDEFGHI (91)
- 8 DISPLAY 3.1111111 (91)
- 9 DISPLAY 44.\$42 (91)

Casos de teste

- 10 DISPLAY 100.\$\$\$\$\$\$\$\$ (91)
- 11 DISPLAY 10000000-M (91)
- 12 DISPLAY FF-8000 (92)
- 13 DISPLAY FFF.7001 (92)
- 14 DISPLAY 8000-END (92)
- 15 DISPLAY 8000-8001 (92)
- 16 DISPLAY AA-A9 (93)
- 17 DISPLAY 7000.0 (93)
- 18 DISPLAY 7FF9-END (94, 97)

Casos de teste

- 19 DISPLAY 1 (94, 97)
- 20 DISPLAY 21-29 (94, 97)
- 21 DISPLAY 4021.A (94, 97)
- 22 DISPLAY -END (94, 96)
- 23 DISPLAY (94, 96)
- 24 DISPLAY -F (94, 96)
- 25 DISPLAY .E (94, 96)
- 26 DISPLAY 7FF8-END (94, 96)
- 27 DISPLAY 6000 (94, 96)

Casos de teste

- 28 DISPLAY A0-A4 (94, 96)
- 29 DISPLAY 20.8 (94, 96)
- 30 DISPLAY 7001-END (95, 97)
- 31 DISPLAY 5-15 (95, 97)
- 32 DISPLAY 4FF.100 (95, 97)
- 33 DISPLAY -END (95, 96)
- 34 DISPLAY -20 (95, 96)
- 35 DISPLAY .11 (95, 96)
- 36 DISPLAY 7000-END (95, 96)

Casos de teste

- 37 DISPLAY 4-14 (95, 96)
- 38 DISPLAY 500.11 (95, 96)
- Ver mais comentários no livro

Exercício – AVL e GCE

- O programa Cal deve exibir para o usuário o calendário de um determinado mês ou de um ano todo. Ele pode receber até dois argumentos. Se mais do que dois argumentos forem dados, os excedentes são simplesmente ignorados. Se dois argumentos forem fornecidos, o primeiro é o mês e o segundo o ano, e o programa deve exibir o calendário do mês solicitado. Se apenas um argumento for fornecido então ele representa um ano e o programa deve exibir o calendário daquele ano todo. Se nenhum argumento for fornecido, deve apresentar o calendário do mês corrente. O valor permitido para o ano é de 1 a 9999. Valores fora do intervalo válido ou argumentos não numéricos devem ser identificados e uma mensagem de erro deve ser dada.

Exercício – AVL e GCE

- No ano de 1752 houve a troca do calendário Juliano para o Gregoriano. Nesse ano houve um ajuste e no mês de setembro não existem os dias 3 a 13. Ou seja, temos o seguinte calendário para aquele mês:

setembro 1752

Do	Se	Te	Qu	Qu	Se	Sá
		1	2	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Além disso, a partir daquele ano estabeleceram-se as atuais regras para definir quando um ano é bissexto. Antes disso, todos os anos múltiplos de 4 eram bissextos.