

1. Publicação nº <i>INPE-3718-PRE/854</i>	2. Versão	3. Data <i>Novembro 1985</i>	5. Distribuição <input type="checkbox"/> Interna <input checked="" type="checkbox"/> Externa <input type="checkbox"/> Restrita
4. Origem <i>DIN/DCS</i>	Programa <i>DENUME</i>		
6. Palavras chaves - selecionadas pelo(s) autor(es) <i>FATORAÇÃO MANIPULAÇÃO ALGÉBRICA LINGUAGEM DE MANIPULAÇÃO</i> <i>DERIVAÇÃO FORMATAÇÃO DE GRÁFICOS</i>			
7. C.D.U.: <i>519.674</i>			
8. Título <i>INPE-3718-PRE/854</i>		10. Páginas: <i>18</i>	
<i>UM MANIPULADOR ALGÉBRICO PARA MICROCOMPUTADORES</i>		11. Última página: <i>A.5</i>	
9. Autoria <i>Vera Helena de Ávila Duarte</i>		12. Revisada por <i>L. A. V. D.</i> <i>Luiz A. Vieira Dias</i>	
Assinatura responsável <i>Vera Helena Duarte</i>		13. Autorizada por <i>M. A. Raupp</i> <i>Marco Antonio Raupp</i> <i>Diretor Geral</i>	
14. Resumo/Notas <i>O sistema a ser apresentado é um "software" de manipulação algébrica de fórmulas, desenvolvido para microcomputadores. Entre as manipulações algébricas executadas pelo sistema estão a derivação, tabulação, fatoraço e formatação de gráficos.</i>			
15. Observações <i>Trabalho apresentado no 2º Encontro Regional de Matemática Aplicada e Computacional realizado no INPE, nos dias 16 e 17 de maio de 1985.</i>			

UM MANIPULADOR ALGÉBRICO PARA MICROCOMPUTADORES

Vera Helena de Ávila Duarte*

Instituto de Pesquisas Espaciais - INPE
Conselho Nacional de Desenvolvimento Científico e
Tecnológico - CNPq
C.P. 515 - 12200 - São José dos Campos - SP - Brasil

RESUMO

O sistema a ser apresentado é um "software" de manipulação algébrica de fórmulas, desenvolvido para microcomputadores. Entre as manipulações algébricas executadas pelo sistema estão a derivação, tabulação, fatoração e formatação de gráficos.

ABSTRACT

The system to be presented is a software for algebraic manipulation of formulae, developed for microcomputers. Among the algebraic manipulations executed by the system are derivation, tabulation, factoring and plotting.

* Mestranda em Computação Aplicada, com bolsa da CAPES.

1 - INTRODUÇÃO

Este sistema é um "software" de manipulação algébrica de fórmulas, desenvolvido na Universidade de Brasília, no Departamento de Estatística, para ajudar os alunos e professores que, por meio dos microcomputadores dos seus departamentos, ou mesmo do microcomputador do Departamento de Estatística, queiram resolver tarefas mais simples relacionadas à derivação, fatoração e tabulação de expressões matemáticas e esboço de gráficos de uma determinada função.

O "software" foi desenvolvido para microcomputadores com sistema operacional CP/M. A linguagem adotada para o seu desenvolvimento foi a PASCAL (PASCAL/M) por ser uma linguagem recursiva, de fácil aprendizado, adequada para operações numéricas e manipulações simbólicas, de fácil estruturação de dados e portátil a qualquer microcomputador com sistema operacional CP/M.

2 - COMPONENTES DO SISTEMA

O sistema é um "software" formado por um conjunto de sub-rotinas inter-relacionadas para execução de algumas manipulações algébricas, sistema este que possui estruturas dendríticas como representação interna das expressões algébricas.

Dentre o conjunto de sub-rotinas podem-se destacar oito módulos principais:

- a) conversação;
- b) analisador;
- c) manipulação de árvores;
- d) manipulação de tabelas;
- e) derivação;
- f) tabulação;
- g) gráfico;
- h) fatoração.

A seguir, serão apresentadas as funções de cada módulo e a forma de relacionamento entre eles.

2.1 - COMPONENTES DO SISTEMA

2.1.1 - MÓDULO DE CONVERSAÇÃO

Este módulo é a interface do sistema com o usuário. É por meio dele que o usuário fará as manipulações que desejar.

A comunicação sistema/usuário pode ser feita utilizando uma linguagem de manipulação, especialmente desenvolvida para esse fim, ou de forma conversacional, para usuários ainda não familiarizados com a linguagem. Existe um intercâmbio entre a linguagem e o "modo conversacional", ou seja, o usuário pode estar no "modo linguagem" e passar para o "modo conversacional", retornando, posteriormente, ao primeiro. Esse módulo é o que gerencia as chamadas de todos os outros módulos.

A linguagem de interação com o usuário foi implementada pela utilização da abordagem "top down" e do modelo recursivo descendente (Gries, 1971) de análise sintática, razão pela qual a adoção de uma linguagem recursiva tornou-se fundamental. Tal linguagem foi construída deterministicamente, admitindo analisador sintático com apenas um símbolo terminal como "lookahead" (gramática LL(1)) (Aho and Ulman, 1972; Hopcroft and Ulman, 1969; Martin, 1972).

A linguagem será apresentada resumidamente através da exemplificação de algumas transações.

1) SOS

O "pedido de socorro" faz com que apareça um cardápio explicando ao usuário quais as funções do sistema e as formas de interação com ele. Isto é, o usuário pode escolher entre o "modo linguagem" e o "modo conversacional".

2) ?

O símbolo "?" significa a passagem do "modo linguagem" para o "modo conversacional". O sistema pedirá os parâmetros para que o usuário entre com a expressão algébrica, ou mostrará as manipulações disponíveis no sistema, se o usuário já tiver fornecido a expressão algébrica.

3) $F(Y)=EXP(5*Y)+RQ(X/3);$

O usuário define sua expressão utilizando as funções exponencial e raiz quadrada, no caso acima. Veja no Apêndice A, Tabela A.1, os operadores permitidos nas expressões. Aqui têm-se "Y" como variável e "X" como constante.

4) $F=12.3E-5*LN(X);$

Quando a variável não for especificada, o sistema assumirá "X". A função utilizada foi o logaritmo neperiano.

5) DERIVE ? #

Passa para o "modo conversacional" onde o sistema pedirá os parâmetros da derivação, ou seja, a ordem da derivada em relação a que variável deverá ser calculada a derivada, bem como derivadas a serem mostradas na tela.

6) DER (3) #

Deriva a expressão F até ordem 3 em relação à variável "X", mostrando a derivada de ordem 3.

7) DERI (4;1,2) #

Deriva a expressão F até ordem 4 em relação à variável "X", mostrando na tela as derivadas de ordem 1, 2 e 4.

8) DER (8; 2:4, 6;Y)#

Deriva a expressão F até ordem 8 em relação à variável "Y", mostrando na tela as derivadas de ordem 2, 3, 4, 6 e 8.

9) TABULE ? #

Passa para o "modo conversacional" onde o sistema pedirá ao usuário os parâmetros da tabulação, ou seja, os valores inicial, final e incremento para a variável e os valores para as constantes, se existirem.

10) TABUL F VI=1, VF=12, INC=2; LC #

Tabula a expressão F com os valores inicial, final e incremento iguais 1, 12 e 12, respectivamente. O comando "LC" faz com que todas as constantes da expressão F sejam listadas para que seja atribuído a elas algum valor.

11) TAB D(3) VF=15, VI=3 #

Tabule a derivada de ordem 3 em relação à variável "X". Quando o valor para o incremento não for especificado, será considerado igual a 1. O sistema listará todas as constantes que ainda não possuem algum valor.

12) TAB D(1;Y) 1,8,2; A=B=5, Z=2 #

Tabula a derivada de ordem 1 em relação à variável "Y". Os valores inicial, final e incremento são 1, 8 e 2, respectivamente. As constantes A e B receberão o valor 5; Z receberá o valor 2.

13) FATORE ? #

Passa para o "modo conversacional" onde o sistema pedirá ao usuário a expressão a ser fatorada.

14) FATOR F #

Fatora a expressão F.

15) FATO D (4;Y) #

Fatora a derivada de ordem 4 em relação à variável "Y".

16) FAT D(2) #

Fatora a derivada de ordem 2 em relação à variável "X".

17) GRÁFICO ? #

Passa para o "modo conversacional" onde o sistema pedirá os parâmetros do gráfico, ou seja, a expressão a ser esboçada na forma de gráfico, os valores para a variável e os valores de mínimo e máximo para o eixo das ordenadas.

18) GRAFI D(3;Y) #

Traça o gráfico da derivada de ordem 3 em relação à variável "Y". Essa expressão já deve ter sido tabulada para obter os valores de "Y" e os valores para o eixo das ordenadas.

19) GRAF d(2) YMAX=50, YMIN=1 #

Traça o gráfico da derivada de ordem 2 em relação à variável "X", que já deve ter sido tabulada para obter os valores de "X" e das constantes, se existirem. Admitiram-se os valores de máximo e mínimo para o eixo das ordenadas, que são 50 e 1, respectivamente.

20) GRA F 1,9; YMIN=0, YMAX=30 #

Traça o gráfico da expressão F, tabulando-o com os valores inicial, final e incremento iguais a 1, 9 e 1, respectivamente. Admitiram-se os valores de mínimo e máximo para o eixo das ordenadas, que são 0 e 30, respectivamente.

21) DESTRUA ? #

Passa para o "modo conversacional" onde o sistema pedirá ao usuário que informe a expressão ou as expressões a serem destruídas.

22) DEST * #

Destrói todas as expressões.

23) DESTR D(*) #

Destrói todas as expressões que foram derivadas em relação à variável "X".

24) DES F, D(3;Y) #

Destrói a expressão F e a derivada de ordem 3 em relação à variável "Y".

25) FIM #

Reinicia a sessão esperando nova expressão para manipular, o que causa a destruição de todas as expressões anteriores.

26) TCHAU #

Encerra a sessão.

2.1.2 - MÓDULO ANALISADOR

Esse módulo é responsável pela análise léxica, sintática e se mântica dos comandos da linguagem de manipulação e, caso necessário, envia mensagens de erro.

O módulo analisador utiliza-se dos módulos de manipulação de árvores (parte semântica) e de tabelas.

O módulo de manipulação de árvores é utilizado, pois, a medida que o analisador reconhece a expressão algébrica, ele constrói a árvore binária equivalente à expressão digitada, já que as árvores binárias são uma forma de representar internamente as expressões em computador. A montagem das árvores será detalhada na Seção 2.1.3.

O módulo de manipulação de tabelas é importante para o reconhecimento dos símbolos terminais durante a análise léxica.

2.1.3 - MÓDULO DE MANIPULAÇÃO DE ÁRVORES

Esse módulo contém as rotinas de percorrimto de árvores, gerenciamento de espaço (alocação e liberação de nós) e construção das árvores (Knuth, 1973).

A representação interna das expressões é estruturada em forma de árvores, ou em forma dendrítica. As árvores são representadas por meio de suas árvores binárias equivalentes, costuradas à direita.

Sua implementação é baseada na estrutura da Figura A.1, Apêndice A, onde:

- LLINK, RLINK são os ponteiros esquerdo e direito do nó. O ponteiro esquerdo aponta para o seu "FILHO" e o direito para o seu "IRMÃO".
- RTAG: "flag" para indicar se o nó está costurado ou não.

- TIPO é o número que indica o tipo de informação contida no nó, ou seja, se é um operador (e qual é esse operador), um número ou um identificador.
- GRAU é o número que indica o grau do operador, e se é um operador binário ou unário.
- PINFO é o endereço em uma tabela de operadores ou de "strings" onde estará a informação. Tal técnica foi adotada para manter a árvore binária com nós de tamanho constante.
- NÁRIO é o número que indica quantos "FILHOS" tem o nó (será exemplificado na Seção 2.1.8).

O módulo de manipulação de árvores está relacionado com os módulos analisador, manipulação de tabelas, derivação e fatoração.

A gerência do espaço é feita alocando um nó quando necessário, ou devolvendo os nós das árvores que não serão mais utilizadas. Com o comando de destruição, todos os nós da árvore especificada pelo comando são enviados a uma lista do espaço disponível para serem reutilizados.

Como já foi mencionado, ao digitar uma expressão algébrica, o analisador reconhece os símbolos e mostra a árvore binária equivalente à expressão digitada. Para isso, utiliza-se uma rotina de "construção de árvores" que compõe novas árvores por meio da junção de outras menores (Knuth, 1973). Seja "x" a informação de um nó e P1 e P2 os ponteiros para as árvores; então:

- a) TREE1(x) constrói uma nova árvore com x como um nó raiz.
- b) TREE2(x,P1) constrói uma nova árvore com x como nó raiz e com P1 como subárvore.
- c) TREE3(x,P1,P2) constrói uma nova árvore com x como nó raiz e com P1 e P2 como subárvores.

Veja um exemplo da construção da árvore binária equivalente à expressão $SEN(X^2)$ no Apêndice A, Figura A.2.

2.1.4 - MÓDULO DE MANIPULAÇÃO DE TABELAS

Esse módulo é utilizado por todos os módulos. Ele faz uso de três tabelas: tabela de palavras reservadas, tabelas de operadores (binários e unários) e tabela de identificadores e números.

a) Tabela de Palavras Reservadas (TPR)

É uma tabela sequencial que contém a palavra e um número correspondente a ela; é uma representação interna utilizada pelo módulo analisador (Apêndice A, Tabela A.2a).

b) Tabela de Operadores (TO)

É uma tabela sequencial que contém todos os operadores binários e unários, com seus respectivos símbolos (números), utilizados pelo módulo analisador (Apêndice A, Tabela A.2b).

c) Tabela de Identificadores e Números (TIN)

Essa tabela armazena os identificadores (constantes e variáveis) e os números utilizados nas expressões algébricas (Apêndice A, Tabela A.2c).

Ela é subdividida em:

- INFOSTR: é uma "string" de 255 posições utilizada para armazenar os números e os identificadores. Pode haver uma ou mais INFOSTR, uma vez que ela é alocada dinamicamente.

- TABCTE:
 - POSI : é a posição na INFOSTR onde começa a "string".
 - TAM : é o tamanho da "string".
 - OFFSET : indica se a INFOSTR é a primeira, ou a segunda, etc.
 - SETNUM : \emptyset (identificador);
1 (número).
 - VALOR : contém o valor real do número. Para os identificadores o campo é \emptyset .

2.1.5 - MÓDULO DE DERIVAÇÃO

Esse módulo é responsável pela derivação das expressões, devendo ser especificadas a ordem da derivada de maior ordem, quais as derivadas que serão mostradas na tela e em relação a que variável devem ser calculadas as derivadas. Nele estão as rotinas de cópia de árvores, de regras de derivação e a de derivação propriamente dita (Knuth, 1973).

O módulo de derivação utiliza-se do módulo de manipulação de árvores e de tabelas, pois o algoritmo de derivação, dada uma determinada árvore equivalente a uma expressão algébrica, gera uma outra árvore que é aquela correspondente à derivada dessa expressão. Isso torna necessários o gerenciamento de espaço e as rotinas de construção de árvores descritas anteriormente.

Para calcular a derivada de ordem N de uma expressão, basta aplicar N vezes o algoritmo de derivação na árvore que for sendo formada (derivada da árvore anterior). A medida que essas árvores são geradas e não há mais necessidade delas, por não serem mostradas na tela, os nós dessas árvores são liberados para a lista do espaço disponível para ser posteriormente alocados por ocasião da construção de outras árvores.

O controle de liberação das árvores derivadas é feito por meio de uma lista encadeada que aponta para a cabeça da lista de cada árvore e contém a ordem de derivação daquela árvore e a variável em relação a qual a expressão foi derivada. Veja um exemplo no Apêndice A, Figura A.3 para a expressão $\text{SEN}(X)$, dado o comando: `DERIVE (3;1)#`.

2.1.6 - MÓDULO DE TABULAÇÃO

Esse módulo executa a tabulação da expressão algébrica, devendo ser fornecidos a ele como parâmetros o valor inicial, final e o incremento para a variável e o valor das constantes, caso existam. Para tanto, ele faz uso do módulo de manipulação de tabelas (TIN), armazenando nela o valor correspondente a cada constante.

2.1.7 - MÓDULO DE GRÁFICOS

Esse módulo é responsável pelo esboço dos gráficos. Como na direção horizontal da tela têm-se 80 posições e na vertical apenas 24, o gráfico terá o seu eixo das abcissas na vertical e o eixo das ordenadas na horizontal, pois há necessidade de maior precisão no eixo das ordenadas.

O sistema oferece ao usuário a possibilidade de escolher os valores mínimo e máximo para o eixo das ordenadas e abcissas. Se o usuário não fornecer algum desses valores, o sistema considerará como valores máximo e mínimo os valores calculados pela tabulação. Estes valores são usados na normalização da função para enquadramento na tela.

O módulo de gráfico utiliza-se dos módulos de tabulação e manipulação de tabelas.

2.1.8 - MÓDULO DE FATORAÇÃO

Esse módulo é responsável pela simplificação das expressões algébricas, utilizando para isso os módulos de manipulação de tabelas e árvores.

A primeira fase da simplificação das expressões consistirá em transformar as árvores com os operadores binários de subtração, divisão e exponenciação em operadores n-ários em termos de somas e multiplicações, executando, em paralelo, as operações com os números e a ordenação destes e das constantes, como mostra o exemplo do Apêndice A, Figura A.4.

Após a transformação dos operadores binários em operadores n-ários, deve-se percorrer a árvore para encontrar subárvores equivalentes para, então, efetuar a simplificação (Apêndice A, Figura A.5). Após a simplificação, a árvore não simplificada é eliminada.

3 - CONSIDERAÇÕES FINAIS

Este "software" foi o primeiro sistema de manipulação simbólica desenvolvido na Universidade de Brasília, estando parcialmente implementado no microcomputador Polymax-201/DP do Departamento de Estatística.

Apesar de executar apenas algumas manipulações mais simples, seu desenvolvimento foi feito de forma modular para que possa ser expandido facilmente, sem prejuízo dos módulos já existentes.

4 - REFERÊNCIAS BIBLIOGRÁFICAS

AHO, A.; ULLMAN, J. *The theory of parsing, translation and compiling*. Englewood Cliffs, NJ, Prentice-Hall, 1972. v. 1.

GRIES, D. *Compiler construction for digital computers*. New York, NY, John Wiley, 1971.

HOPCROFT, J.; ULLMAN, J. *Formal languages and their related automata*. Addison-Wesley, 1969.

KNUTH, D. *The Art of computer programming - fundamental algorithms*. Massachusetts, Addison-Wesley, 1973.

MARTIN, D. Formal languages and their related automata. IN: Cardenas, A.; Presser, L.; Marin, M. (ed.). *Computer science*. New York, John Wiley, 1972.

APÊNDICE A

ILUSTRAÇÕES

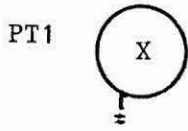
TABELA A.1

OPERADORES PERMITIDOS

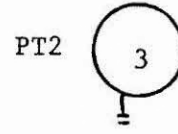
OPERADORES BINÁRIOS		OPERADORES UNÁRIOS	
SÍMBOLO	SIGNIFICADO	SÍMBOLO	SIGNIFICADO
+	soma	SEN	seno
-	subtração	COS	co-seno
*	multiplicação	TG	tangente
/	divisão	COTG	co-tangente
∞	exponenciação	SEC	secante
		CSEC	co-secante
		LOG	logaritmo base 10
		LN	logaritmo neperiano
		RQ	raiz quadrada
		QUAD	quadrado
		EXP	exponencial
		LAMB	lambda
		DELTA	delta

PINFO	LLINK	RLINK	RTAG	TIPO	GRAU	NÁRIO
-------	-------	-------	------	------	------	-------

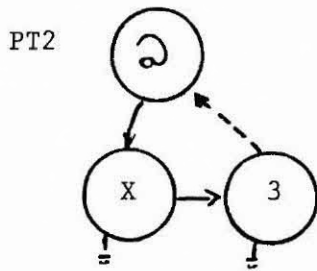
Fig. A.1 - Representação do nó da árvore binária equivalente.



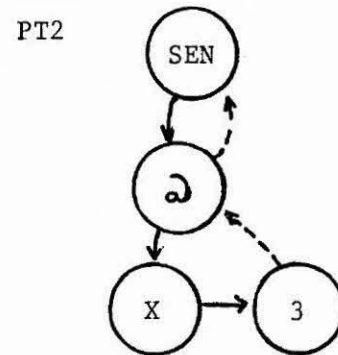
PT1 := TREE(X)
(a1)



PT2 := TREE(3)
(a2)

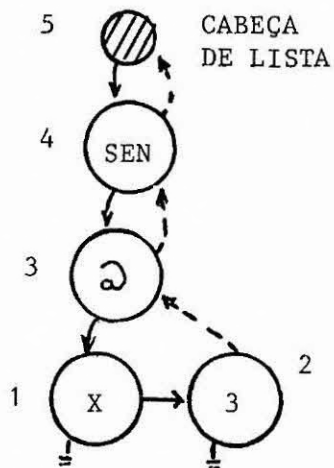


PT2 := TREE3(2, PT1, PT2)
(a3)



PT2 := TREE2(SEN, PT2);
(a4)

(a) Construção da árvore binária



(b) Árvore binária

1	1	0	2	0	19	0	0
2	2	0	3	1	50	0	0
3	19	1	4	1	18	2	2
4	4	3	5	1	3	1	1

(c) Estrutura dos nós

Fig. A.2 - Representação da expressão SEN(X3) na sua árvore binária equivalente.

TABELA A.2

PALAVRAS RESERVADAS (TPR), OPERADORES (TO)
E IDENTIFICADORES E NÚMEROS (TIN)

D	29
F	33
LC	41
VI	42
VF	43
SOS	45
FIM	34
INC	44
TCHAU	40
LISTE	25
YMAX	32
YMIN	31
DERIVE	35
FATORE	38
TABULE	36
GRÁFICO	37
DESTRUA	39
CONSTANTE	46

(a) TPR

TG	0
RQ	1
COS	2
SEN	3
LN	4
EXP	5
COTG	6
QUAD	7
DELTA	8
LAMB	9
SEC	11
CSEC	12
LOG	13
+	14
-	15
*	16
/	17
∞	18

(b) TO

	POSI	TAM	OFFSET	SETNUM	VALOR
1	1	1	1	0	0
2	2	1	1	1	2
.					
.					
.					
50					

(c1) TABCTE

X	2	. . .	
---	---	-------	--

1 2

(c2) INFOSTR

(c) TIN

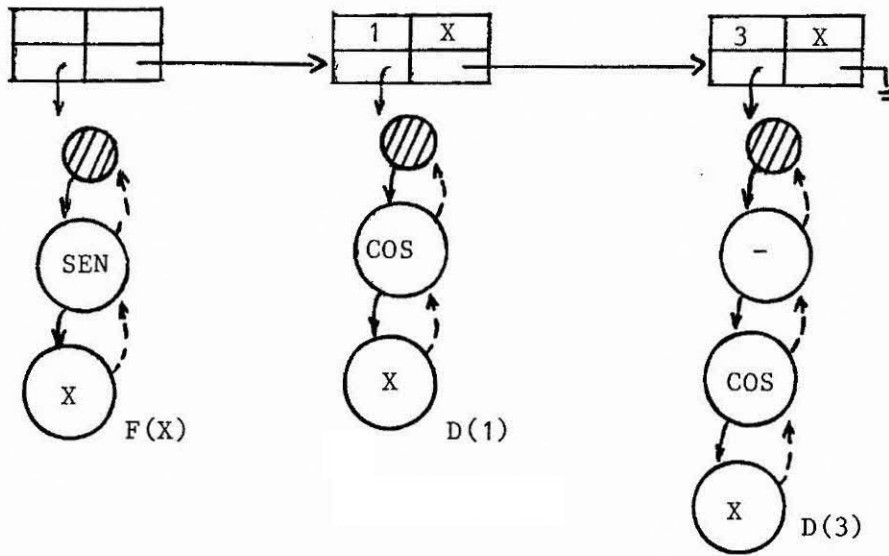


Fig. A.3 - Árvores derivadas encadeadas.

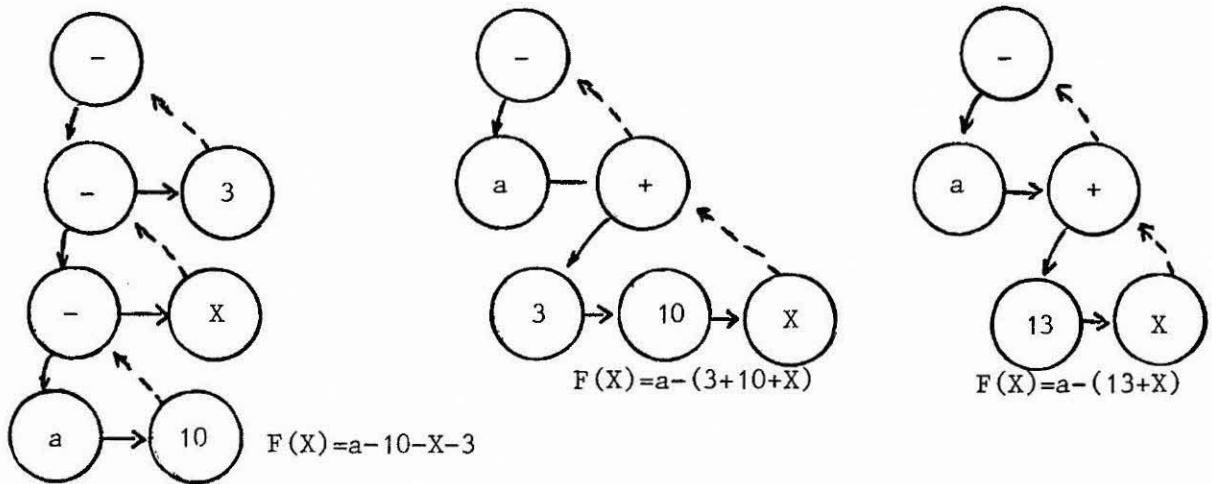


Fig. A.4 - Transformação da árvore com os operadores binários de subtração em operador de soma.

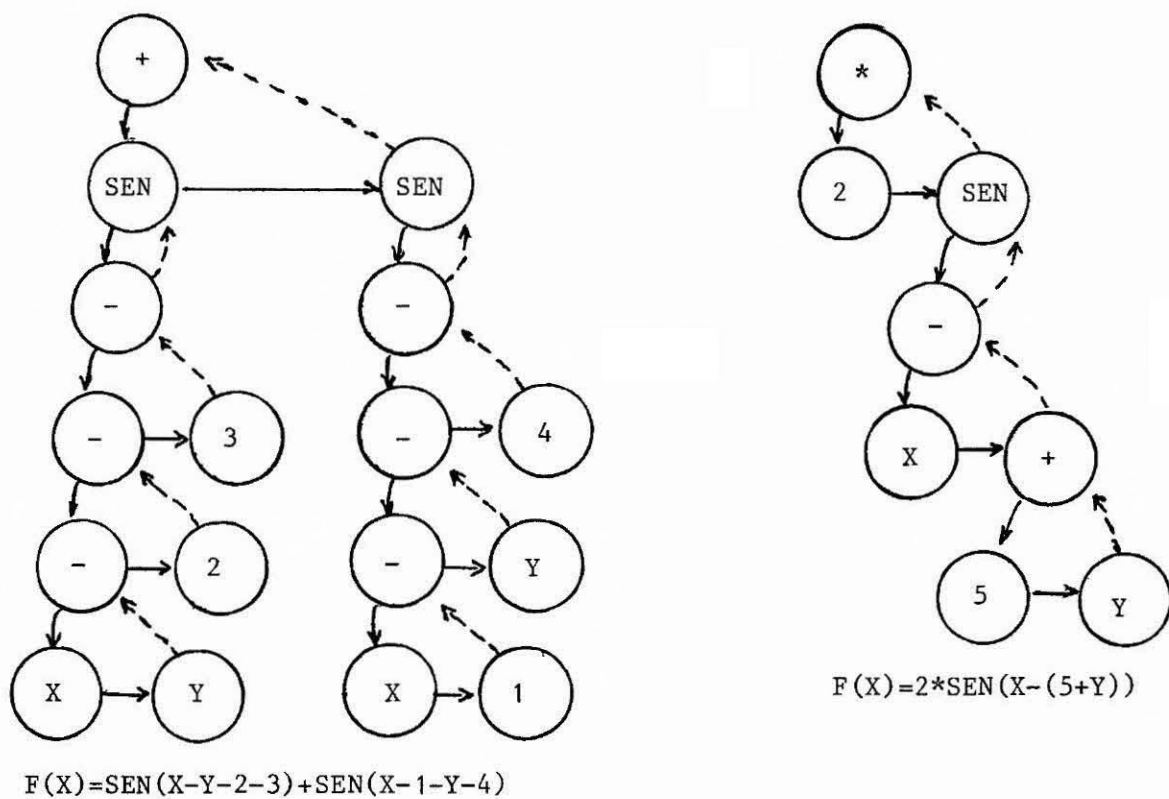


Fig. A.5 - Simplificação da expressão $\text{SEN}(X - Y - 2 - 3) + \text{SEN}(X - 1 - Y - 4)$.