

# A 1. FORTRAN プログラムにおける誤まりの傾向について

五十嵐 滋 (東京大学)

東京大学大型計算機センターでは、一般利用者がFORTRANにより作製し計算依頼したプログラムの計算結果から適当な抽出を行って、数千件を資料として保存しているが、その内約半数はいわゆるデバック段階のもので何らかの誤まりを含んでいる。その一部分を対象に、これらの誤まりについて多少の統計的調査を行った結果に基づき、すでに初心者や不注意者のおかし易い誤まりを避けるためのチェック・リストを作製し使用に供している。しかし誤まりを避け、また誤まりの診断を容易にするにはFORTRAN文法自体の問題、コンパイラを含むシステム全体の問題等多々あることが気付かれた。これらについて述べる。

## 1 コンパイル時に検出される誤まりの分布

昭和41年4月～7月の一般依頼計算から、1169件のジョブをほぼ公平に抽出して、コンパイル時に出されたエラー・メッセージの度数分布を整理して示すと、表1のようになる。ただしエラー・メッセージの番号はHITAC5020コンパイラ固有のものである。同一ジョブで、同じエラーを多数回出しても1回と数えてある。たとえば見出し1のところでは、全体の度数が19になっていることは、エラー番号3のエラー・メッセージを少なくとも1個含んでいたジョブが19個あったことを示す。その右の数字は、上の意味でのエラーの出現総数1315に対する19の百分率である。

表1で見られるように、各組の度数分布は互いにかなりよく似ているので、エラー番号の分布はかなり安定しているといえる。

エラー番号	エラー名	出現総数	百分率 (%)
1	...	1315	100.0
2	...	...	...
3	...	...	...
4	...	...	...
5	...	...	...
6	...	...	...
7	...	...	...
8	...	...	...
9	...	...	...
10	...	...	...
11	...	...	...
12	...	...	...
13	...	...	...
14	...	...	...
15	...	...	...
16	...	...	...
17	...	...	...
18	...	...	...
19	...	...	...
20	...	...	...
21	...	...	...
22	...	...	...
23	...	...	...
24	...	...	...
25	...	...	...
26	...	...	...
27	...	...	...
28	...	...	...
29	...	...	...
30	...	...	...
31	...	...	...
32	...	...	...
33	...	...	...
34	...	...	...
35	...	...	...
36	...	...	...
37	...	...	...
38	...	...	...
39	...	...	...
40	...	...	...
41	...	...	...
42	...	...	...
43	...	...	...
44	...	...	...
45	...	...	...
46	...	...	...
47	...	...	...
48	...	...	...
49	...	...	...
50	...	...	...
51	...	...	...
52	...	...	...
53	...	...	...
54	...	...	...
55	...	...	...
56	...	...	...
57	...	...	...
58	...	...	...
59	...	...	...
60	...	...	...
61	...	...	...
62	...	...	...
63	...	...	...
64	...	...	...
65	...	...	...
66	...	...	...
67	...	...	...
68	...	...	...
69	...	...	...
70	...	...	...
71	...	...	...
72	...	...	...
73	...	...	...
74	...	...	...
75	...	...	...
76	...	...	...
77	...	...	...
78	...	...	...
79	...	...	...
80	...	...	...
81	...	...	...
82	...	...	...
83	...	...	...
84	...	...	...
85	...	...	...
86	...	...	...
87	...	...	...
88	...	...	...
89	...	...	...
90	...	...	...
91	...	...	...
92	...	...	...
93	...	...	...
94	...	...	...
95	...	...	...
96	...	...	...
97	...	...	...
98	...	...	...
99	...	...	...
100	...	...	...

表 1

見出し	エラー 番 号	3,4月の 標本(407件)		5,7月の 標本(411件)		6月の 標本(35件)		全 体 (1169件)	
		度数	%	度数	%	度数	%	度数	%
1	3	7	1.43	9	1.75	3	0.96	19	1.44
2	4	11	2.24	10	1.94	4	1.28	25	1.90
3	6	6	1.22	4	0.77	6	1.92	16	1.21
4	11	2	0.40	4	0.77	0	0	6	0.45
5	12	6	1.22	2	0.38	3	0.96	11	0.83
6	21	1	0.20	1	0.19	4	1.28	6	0.45
7	24	3	0.61	3	0.58	0	0	6	0.45
8	27	2	0.40	1	0.19	3	0.96	6	0.45
9	32	3	0.61	6	1.16	3	0.96	12	0.91
10	33	9	1.84	4	0.77	8	2.56	21	1.59
11	36	10	2.04	5	0.97	1	0.32	16	1.21
12	37	1	0.20	3	0.58	2	0.64	6	0.45
13	40	7	1.43	3	0.58	0	0	10	0.75
14	44	12	2.45	5	0.97	2	0.64	19	1.44
15	53	7	1.43	13	2.52	8	2.56	28	2.12
16	64	6	1.22	4	0.77	3	0.96	13	0.98
17	67	6	1.22	3	0.58	3	0.96	12	0.91
18	73	20	4.08	23	4.47	8	2.56	51	3.87
19	74	2	0.40	2	0.38	7	2.24	11	0.83
20	90	5	1.02	5	0.97	3	0.96	13	0.98
21	92	7	1.43	10	1.97	3	0.96	20	1.52
22	95	8	1.63	4	0.77	3	0.96	15	1.14
23	98	4	0.81	4	0.77	3	0.96	11	0.83
24	102	9	1.84	14	2.72	7	2.24	30	2.28
25	103	3	0.61	7	1.36	4	1.28	14	1.06
26	104	3	0.61	3	0.58	2	0.64	8	0.60
27	111	8	1.63	6	1.16	1	0.32	15	1.14
28	116	3	0.61	3	0.58	1	0.32	7	0.53
29	118	3	0.61	2	0.38	1	0.32	6	0.45
30	122	5	1.02	3	0.58	4	1.28	12	0.91
31	134	10	2.04	10	1.94	3	0.96	23	1.74
32	150	5	1.02	3	0.58	0	0	8	0.60
33	156	3	0.61	3	0.58	1	0.32	7	0.53
34	166	12	2.45	22	4.28	7	2.24	41	3.11

番号)	エラーの内容	エラーの レベル	ジョブ1 件に含ま れる割合 (%)	順位
4)	1個のステートメントで右カッコの数が左カッコの数よりも多い。	1	1.62	⑮
		4	2.13	
		1	1.36	
		1	0.51	
		4	0.94	
		4	0.51	
		4	0.51	
		4	0.51	
		1	1.02	
		33)	HARP 5020 で許されていないステートメントが現れた。	
36)	コア変数が使われるべきなのに使われなかつた。	4	1.36	*
		4	0.51	
		2	0.85	
44)	FORMAT ステートメントの仕様の書き方に誤りがある。	2	1.62	⑳
53)	1個のステートメントで左カッコの数が右カッコの数よりも多い。	4	2.39	㉑
		4	1.11	
67)	添字なし変数に左カッコがついている。	4	1.02	*
73)	ステートメントナンバー ..... が未定義である。	2	4.36	㉒
		4	0.94	
		4	1.11	
92)	添字付き変数が添字の情報を持たないで現れた。	4	1.71	㉓
95)	ステートメント関数の定義式に添字付変数が現れた。	4	1.28	*
		1	0.94	
102)	ステートメント番号を2重に定義している。	4	2.56	㉔
		4	1.19	
		4	0.68	
111)	配列名が現われるべきところに現われない。	4	1.28	*
		4	0.59	
		1	0.51	
		4	1.02	
134)	算術ステートメントの左辺の名前に誤りがある。	4	1.96	㉕*
		2	0.68	
166)	ステートメント番号が使われているが、それが定義されていない。	4	0.59	
		4	3.50	㉖

見出し	エラー 番号	3,4月の 標本(407件)		5,7月の 標本(411件)		6月の 標本(351件)		全 体 (1169件)	
		度数	%	度数	%	度数	%	度数	%
35	168	2	0.40	2	0.38	3	0.96	7	0.53
36	172	2	0.40	5	0.97	3	0.96	10	0.75
37	173	8	1.63	7	1.36	5	1.60	20	1.52
38	175	14	2.86	15	2.91	8	2.56	37	2.81
39	181	12	2.45	16	3.11	9	2.88	37	2.81
40	185	1	0.20	7	1.36	0	0	8	0.60
41	192	7	1.43	12	2.33	7	2.24	26	1.97
42	197	11	2.24	10	1.94	9	2.88	30	2.28
43	200	9	9.84	18	3.50	9	2.88	36	2.73
44	201	22	44.9	11	2.14	7	2.24	40	3.04
45	204	3	0.61	2	0.38	2	0.64	7	0.53
46	205	5	1.02	11	2.14	11	3.52	27	2.05
47	206	7	1.43	10	1.94	3	0.96	20	1.52
48	215	21	4.29	22	4.28	20	6.41	63	4.79
49	216	95	19.42	89	17.31	65	20.83	249	18.93
50	217	25	5.11	20	3.89	14	4.48	59	4.48
計		456	93.25	461	89.68	286	91.66	1203	91.48
上記以外の番号 0でないもの: 53件 0のもの: 159件		33	6.75	53	10.31	26	8.33	112	8.51
総 計		489	100.0	514	100.0	312	100.0	1315	100.0

番号)	エラーの内容	エラーの レベル	ジョブ1 件に含ま れる割合 (%)	順位
		1	0.59	
		1	0.85	
173)	D0ループのかさなり方に誤りがある。	4	1.71	(18)
175)	算術式の中に許されない組合せ(二つの演算数の型に関して)がある。	2	3.16	(7)
181)	算術式, 論理式の中で記号が変なところに現われた。例えば先頭に*が	2	3.16	(7)
		6	0.68	
192)	D0の指標がループの中で変えられている。	2	2.22	(14)
197)	このステートメントが実行されることはない。	1	2.56	(10)
200)	このステートメントはFORMATステートメントであるが, それが入出力ステートメントで使われていない。	1	3.07	(9)
201)	このステートメント番号は, 入出力ステートメントに現われているが, このステートメントはFORMATステートメントではない。	2	3.42	(6)
		2	0.59	
205)	最後の実行可能なステートメントがG0 T0型かSTOPステートメントでない。	1	2.30	(15)
206)	D0ループの中にステートメント関数を定義するステートメントがある。	1	1.71	(18)*
215)	変数 [ ] は実行可能なステートメントに現れなかった。	1	5.88	(2)
		1	21.30	(1)
216)	変数 [ ] はこの左辺, READステートメントまたはASSIGNステートメントとのいずれかに現われているがそれを使用しているところがない。	1	5.04	(3)
217)	変数 [ ] の値を使っているところはあるが, その値を定義している所がない。			
			9.58	

## 2 FORTTRANのチェック・リスト

1. のような度数分布の安定性から、FORTTRAN のチェック・リストを作ることが考えられたが、この場合誤りのそもそもの原因を指摘するように注意する必要がある。たとえば配列の宣言を忘れたことが主因となって起るエラーは、表1の\*印のがそれであるが、数種類あるが、これは人間に対しては配列の宣言を忘れていないかと尋ねる方が明らかに望ましい。このような点に留意しながら、比較的初期の統計に基いて作製し、少しずつ改訂しながら使用して来たのが、表2のようなチェック・リストである。これについては或る程度の有効性を、サンプリングによって確かめることができた。しかしながらなお改良すべき点もある。

## チェック・リスト

(HITAC5020FORTRAN用)

1. 算術式の中で実数と整数との四則演算をしていませんか？	1. [誤り例] $X = I * A + 2 * B \dots\dots$ Iとせずに FL $\bar{O}$ AT(I)と2とせずに2.0としなければなりません。
2. IFの中で実数と整数とを比較していませんか？	2. [誤り例] IF(A.GE.O)G $\bar{O}$ T $\bar{O}$ 1..... IF(A.GE.O.O)としなければなりません。
3. 配列はすべて忘れずに宣言してありますか？ 添字なし変数名と重複している配列名はありませんか？	3. 宣言を忘れた時のエラー・メッセージの出した少なくとも5通りありますが、そのどれも誤りを正しく教えてくれません。
4. 添字の形は許されるものだけですか？	4. [誤り列] $X = A(M * N + 1) + B \dots\dots$ 添字は 定数 * N ± 定数の形までしか許されません。
5. 添字の値が実行時に、宣言された範囲からはみ出さずおそれはありますか？ 0もいけません。	5. 配列を持つサブルーチンへの実引数、C $\bar{O}$ - MPUTED G $\bar{O}$ T $\bar{O}$ の添字にも注意して 下さい。この誤りがあるとMEM $\bar{O}$ R $\bar{Y}$ PR- $\bar{O}$ TECT, HALT, S $\bar{O}$ FTWARE INST- RUCTION などの難解なエラーを起し、 またエラー・メッセージさえ出さずに無意味な計算を行わせることになります。
6. G $\bar{O}$ T $\bar{O}$ の行先のステートメント番号は正しい位値についていますか？ 同じステートメント番号を2ヶ所以上に使っていませんか？	6. FORMATステートメント番号とREAD, WRITEステートメントで指定した番号との 対応も確認して下さい。
7. D $\bar{O}$ ループは正しく閉じていますか？ D $\bar{O}$ ループの中でD $\bar{O}$ の指標パラメータを変えていませんか？	7. [誤り例] D $\bar{O}$ 9 I=1,100..... D $\bar{O}$ ループ I=I+1                    D $\bar{O}$ ループ 9C ONTINUE                チンを呼んだり、 G $\bar{O}$ T $\bar{O}$ で一旦ループの外に出て戻 る場合にも注意して 下さい。

<p>8. <b>F<math>\bar{O}</math>R<math>\bar{M}</math>A<math>\bar{T}</math></b> ステートメントの書き方に形式上の誤りはありませんか？ 入出力リストとの対応は完全ですか？</p> <p>9. サブルーチン名や長い変数名に綴りミスはありませんか？</p> <p>10. 語数が計算機の容量を超えるおそれはありませんか？ (東大大型機センターの主システムでは、コアに51K語、ドラムに260K語までしか入りません。)</p> <p>X 穿孔依頼をする時は、1とIとノ、0と0などの区別がパンチャーにはっきり分るかどうか再確認しましょう。</p> <p>Y 計算依頼をする時は、制御カードが正しくそろっているか、角が傷んでいるカードがないかを確認しましょう。</p>	<p>8. たとえばH要素の字数を数え違えていませんか？</p> <p>9. [誤り例] A=-ALOG(X).....ALOGでなくALOGと書いて下さい。</p> <p>10. <b>F<math>\bar{O}</math>R<math>\bar{T}</math>RAN</b> の1ステートメントは平均して5~6語を、1語長配列はその要素の個数だけの語を、2倍長、4倍長の配列はそれぞれの要素の個数の2倍、4倍だけの語を占めます。</p>
--	--

東京大学大型計算機センター・プログラム言語研究室

### 3 コンパイル時に検出されないエラー

HITAC5020 **F $\bar{O}$ R $\bar{T}$ RAN** コンパイラーのコンパイル時に検出されないエラーの種類を列挙すると次のようになる。

- a) **F $\bar{O}$ R $\bar{M}$ A $\bar{T}$**  と入出力リストとの対応。
- b) 外部名の参照(\*)。
- c) 仮引数と実引数との対応。
- d) ライブラリー基礎関数の引数の範囲。
- e) **F $\bar{O}$ R $\bar{M}$ A $\bar{T}$**  のパラメータの誤り。
- f) データ・カードの不足。
- g) **O $\bar{V}$ E $\bar{R}$ F $\bar{L}$ O $\bar{W}$**  と0による除算。
- h) **D $\bar{I}$ M $\bar{E}$ N $\bar{S}$ I $\bar{O}$ N** ステートメントで宣言された範囲をこえて配列を用いること。
- i) 算術式の中で符号を間違えるたぐいのこと。

これらについては表3のような一覧表をつくり、プログラム指導・診断の便に供している。要は出力に現れたことから、入力の誤りを能率的に検出することである。

この内 a), b), c), e) はコンパイル時にか、または遅くともロード時に検出されることが望ましい。d), f), g) は実行時にしか検出されないのもやむを得ない。h) は実行時にも検出されないが、このチェックの機能は、たとえば **D $\bar{E}$ B $\bar{U}$ G** ステートメントの一種としてでも、付加した方がよい。i) については全く別の接近を考えなければならない。

またFORTRAN プログラムそのものが正しくとも、

j) オペレーティング・システムを制御するカードに不備があること

がしばしばある。この誤りは、時には制御カードがプログラム単位の中にまぎれ込んでいたりして、FORTRAN 文法上の多量なミスとして現れる結果になり、エラーの統計を、真相から遠ざけることになる。

これらの点について、42年11月現在、同年10月の一般ジョブ約500件についての実態調査と、チェック・リストの追試を行っている。

	検出の時期と主体	形式上の特徴	実 例	エラーの内容
第 1 類	FORTRAN のソース・プログラムを翻訳する際、コンパイラーが出すもの。(FORTRANマニュアル巻末(英文), またはFORTRANコンパイル時のエラー・メッセージ(邦文)に一覧表あり)	*ソース・プログラムの行間か、ENDステートメントの直後で次の\$カードより前にある。 *メッセージの後にERRORxxx, LEVEL x. (xは数字)の表示がある。	A = A + 1の後に、INVALID TYPE COMBINATION OF TWO OPERANDS IN ARITHMETIC EXPRESSION. ERROR 175, LEVEL 2.	各(主またはサブ)プログラム単位の中で検出できる文法上の誤り。主に各ステートメントの誤り。(FORTRANと入出力リストとの対応はここではチェックされない。)
第 2 類	計算の実行開始直前に、ローダーが出すもの。(オペレーティング・システム・プログラマーズ・マニュアルに一覧表あり)	*\$ DATAまたは\$ EOFより後にあり、計算結果は全く出ていない。 *CARD, DECK, LINK, JOB, CHAINなどの単語を含む。 *末尾にLOADING CONTINUES. . . またはLOADING IS DELETED. と表示されることが多い。 *番地の表示がない。	CALL SUBR というステートメントが現れているジョブの中に、SUBRというサブルーチンの本体となっているデックが入っていないとき、\$ DATA(または\$ EOF)の後に、SOME DECKS WHICH WOULD CONTAIN EXTERNAL NAME SUBR ARE MISSING.	デックの相互関係など、ジョブ全体を調べてはじめて検出できるもの。外部名の参照など。(サブプログラムの仮引数と実引数との対応はここでも、他の所でも全くチェックされない。誤りがあると奇妙な計算を続ける)



	検出の時期と主体	形式上の特徴	実 例	エラーの内容
第 3 類	計算の実行中に、FORTRANのシステム内のサブルーチンが呼び出された際、出すもの。(FORTRANのエラーメッセージに内容が詳述されている)	<ul style="list-style-type: none"> <li>*計算結果の途中にある。</li> <li>*VALUE OF FUNCTION の言葉を含むか、入出力に関する内容である。</li> <li>*末尾に CALL SUBROUTINE <u>ルーチン名</u> AT <u>I</u> <u>内部ステートメント番号</u> の表示がある。</li> </ul>	<p>FORMAT で I 変換が指定されているとき、データ・カードの該当する欄内に小数点や E, D の文字が現われると、その READ ステートメントが実行されようとした時点で</p> <p>THE INPUT DATA IS INVALID.</p> <p>CALL SUBROUTINE NAME H &amp; DRI AT I0019.</p> <p>(注: 内部ステートメント番号 19 に対応する READ ステートメントの場合)</p>	ライブラリー基礎関数の引数の値や、関数値の精度が異常な場合それを指摘する。(例. 負数の SQRT を求めようとした) このようにメッセージが出たときその具体的な処理は当該マニュアルに記述がある。(上例については、絶対値の SQRT が関数値なる) 入力ステートメントで指定された FORMAT と、データ・カードが足りないこと、出力や可変 FORMAT のとき、各要素のパラメータ(例 3F12.4 の 3 種の整数)が異常であること)
第 4 類	実数の加減乗除算、または整数の除算の実行時に、FORTRAN システム・サブルーチンが割り込んで出すもの	<ul style="list-style-type: none"> <li>*計算結果の途中にある。</li> <li>*OVERFLOW または ZERO-DIV の文字の後に、絶対番地の表示がついている。</li> </ul>	<p>OVERFLOW</p> <p>10398</p> <p>または</p> <p>ZERO-DIV.</p> <p>9921</p> <p>など</p>	<p>実数の加減乗除の結果の絶対値が、約 <math>10^{38}</math> を超え、正しく表示できなくなったとき OVERFLOW のメッセージを出す。その演算命令語の所在絶対番地が表示され、結果は約 <math>10^{38}</math> になって、計算が続行される。</p> <p>実数または整数の除算で除数が 0 のと</p>

	検出の時期と主体	形式上の特徴	実 例	エラーの内容
				き, ZERO-DIV. のメッセージを出し, 除算を実行せず(すなわち除数が1と同じ結果になる)先に進む. 表示の意味は上に同じ.
第 5 部	計算時間またはラインプリンターでの印刷行数(またはページ数)が, あらかじめ指定された限界を超えたとき, 計算が打ち切られるとともに, ジョブ・モニターから出されるもの	*計算結果が中断された後, アカウント・データの直前にある. *HIJŌB ERRŌR の語で始り, TIME ŌVER または LINE ŌVER と表示される.	HIJŌB ERRŌR 12. TIME ŌVER. または HIJŌB ERRŌR 16. LINE ŌVER. (注. 12, 16 は不変)	(略)
第 6 部	計算実行中, 異常な現象が発生したとき, 計算が打ち切られるとともに, ジョブ・モニターから出されるもの. (オペレーティング・システム・プログラマーズ・マニュアル巻末に一覧表あり)	上と同様で, 個々のエラーに相当する語が表示される.	HIJŌB ERRŌR 11. MEMŌRY PRŌ-TECT. または HIJŌB ERRŌR 14. SŌFTWARE IN-STRUCŌIŌN. など.	主として, 配列の値を書きかえる演算を実行する際, 添字の値が正整数でなかったり, DIMEN-SIŌN ステートメントで宣言された範囲を超えたことが原因となって(直接的または間接的に, プログラムが脱線しはじめ, ついに明らかに異常な現象が起ってはじめて)出る.
	検出されないもの			実引数, 仮引数の対応や, 添字の実行中の値の異常は, ある場合には上の HIJŌB エラーを引き

検出の時期と主体	形式上の特徴	実 例	エラーの内容
			<p>起して検出されるが、必ずしも常に検出される保証はない。</p> <p>無論算術式の中で符号を間違えてあっても検出はできない（しかし変数名を間違えたことについては、弱いながらもコンパイラやローダに検出機能はある）</p>

本 PDF ファイルは 1968 年発行の「第 9 回プログラミング・シンポジウム報告集」をスキャンし、項目ごとに整理して、情報処理学会電子図書館「情報学広場」に掲載するものです。

この出版物は情報処理学会への著作権譲渡がなされていませんが、情報処理学会公式 Web サイトの [https://www.ipsj.or.jp/topics/Past\\_reports.html](https://www.ipsj.or.jp/topics/Past_reports.html) に下記「過去のプログラミング・シンポジウム報告集の利用許諾について」を掲載して、権利者の検索をおこないました。そのうえで同意をいただいたもの、お申し出のなかったものを掲載しています。

#### 過去のプログラミング・シンポジウム報告集の利用許諾について

情報処理学会発行の出版物著作権は平成 12 年から情報処理学会著作権規程に従い、学会に帰属することになっています。

プログラミング・シンポジウムの報告集は、情報処理学会と設立の事情が異なるため、この改訂がシンポジウム内部で徹底しておらず、情報処理学会の他の出版物が情報学広場 (=情報処理学会電子図書館) で公開されているにも拘らず、古い報告集には公開されていないものが少からずありました。

プログラミング・シンポジウムは昭和 59 年に情報処理学会の一部門になりましたが、それ以前の報告集も含め、この度学会の他の出版物と同様の扱いにしたいと考えます。過去のすべての報告集の論文について、著作権者（論文を執筆された故人の相続人）を探し出して利用許諾に関する同意を頂くことは困難ですので、一定期間の権利者搜索の努力をしたうえで、著作権者が見つからない場合も論文を情報学広場に掲載させていただきたいと思います。その後、著作権者が発見され、情報学広場への掲載の継続に同意が得られなかった場合には、当該論文については、掲載を停止致します。

この措置にご意見のある方は、プログラミング・シンポジウムの辻尚史運営委員長 ([tsuji@math.s.chiba-u.ac.jp](mailto:tsuji@math.s.chiba-u.ac.jp)) までお申し出ください。

加えて、著作権者について情報をお持ちの方は事務局まで情報をお寄せくださいますようお願い申し上げます。

期間：2020 年 12 月 18 日～2021 年 3 月 19 日

掲載日：2020 年 12 月 18 日

プログラミング・シンポジウム委員会

情報処理学会著作権規程

<https://www.ipsj.or.jp/copyright/ronbun/copyright.html>