

Univesidad de Buenos Aires - FIUBA  
66:20 Organización de Computadoras  
Trabajo práctico 0: Infraestructura básica

Barrera Oro, Rafael (83240)  
Bacigaluppo, Ivan (98064)  
Irrazabal, Diego (98125)

26 de Septiembre del 2017

# Índice

<b>1. Documentación</b>	<b>2</b>
1.1. Diseño . . . . .	2
1.2. Implementación . . . . .	4
1.2.1. void print_version() . . . . .	4
1.2.2. void print_usage() . . . . .	4
1.2.3. int is_pal(char* word) . . . . .	4
1.2.4. void chomp(char* word) . . . . .	4
1.2.5. void handle_line(char* line, FILE* out_f) . . . . .	4
1.2.6. void handle(char* input_file_name, char* output_file_name)	4
1.2.7. int main(int argc, char** argv) . . . . .	4
1.3. Compilación . . . . .	4
1.4. Ejecución . . . . .	5
<b>2. Casos de prueba</b>	<b>6</b>
2.1. Makefile . . . . .	6
2.2. Validación de parámetros . . . . .	6
2.2.1. Input . . . . .	6
2.2.2. Output . . . . .	6
2.3. Utilizado entrada y salida standard . . . . .	6
2.4. Utilizado archivos . . . . .	6
<b>3. Código fuente</b>	<b>7</b>
3.1. C . . . . .	7
3.2. Assembly (MIPS) . . . . .	12
<b>4. Conclusiones</b>	<b>70</b>

# 1. Documentación

## 1.1. Diseño

Este primer tp es bastante simple puesto que el objetivo real no es el desafío de implementarlo si no familiarizarnos con las herramientas nuevas que se utilizarán a lo largo del curso.

Comenzamos realizando un archivo makefile bastante simple, con algunos flags, para poder compilarlo y ejecutarlo en cuanto antes. Para así poder empezar a probar cualquier cambio que hicieramos, y asegurarnos que compilara adecuadamente con el comando make. Este makefile compilaría el código mediante los siguientes comandos:

```
gcc -g -Wall -o tp0 tp0.c
```

Luego comenzamos con el código ya propio de lo pedido en la consigna. En primer lugar, armamos el main de manera que distinguiese que opción se había ingresado para ejecutar el programa. Al verificar esto, armamos una función para el comando -h, la cual imprime lo pedido; con el comando -V directamente la versión y, por último, hay una función encargada de verificar cual es el archivo de entrada y cual el de salida del programa. En los primeros dos casos el programa termina inmediatamente después de la impresión.

La parte importante del programa fue implementada en la función handle que recibe los argumentos correspondientes como parámetros si los hubiese, o NULL en caso contrario. Esta función se encarga de procesar los archivos de entrada y salida, o asignar los input o output standard para trabajar si alguno hubiera sido pasado como NULL.

Siguiendo el modelo de implementación incremental, en una primera básica versión esta función requeriría un archivo input existente, e imprimiría el resultado por la consola. De esta forma se simplifica la verificación del correcto funcionamiento del programa.

Finalmente, handle realiza la verificación del archivo de entrada. Si lo recibido para este mismo es distinto de NULL, lo abre para lectura; en caso contrario, toma como entrada por defecto la terminal (stdin). Luego, de manera similar, verifica el archivo de salida. Si es distinto de NULL, lo abre pero para escritura en este caso y, de manera contraria, si no se le pasó por comando de entrada, utiliza por defecto la terminal como salida (stdout). Cualquier error en la apertura de los archivos se informa mediante la salida estándar stderr.

A continuación se pasó a procesar el archivo de entrada, leyendo palabra por palabra todavía con la función fscanf utilizando un buffer de 30 chars, que sería la longitud máxima de cada palabra. Estas palabras eran luego pasadas por parámetro a una función es\_palindromo que retornaría un entero booleano, 1 para true y 0 para false, de acuerdo si la palabra era efectivamente un palíndromo. En primer lugar, solo se indicó que retornara siempre un mismo valor 1, de forma que verifiquemos que correctamente se impriman las palabras procesadas del archivo.

El procesado de palabras fue el primer problema encontrado, puesto que la aplicación debería ser indiferente ante la longitud de estas y por lo tanto el buffer de 30 char y fgets no era utilizables al final del día. Este problema se solucionó mediante el uso de la función getline antes de guardar una línea del archivo determina su longitud para el buffer. Así después se procesan las palabras dentro de este buffer.

Luego nos encontramos con la única parte del código no trivial, que era la implementación de la función que determine si la palabra era o no un palíndromo. Leyendo la base del algoritmo, determinamos que la mejor forma de implementarlo era con un ciclo basado en dos puntas, una al principio y una al final de cada palabra, que la recorran hasta encontrarse en el medio, verificando ser iguales en todo momento. Así, el ciclo se rompe si estas puntas son distintas, lo que significa que la palabra no sería un palíndromo.

Ya implementado y probado esto, se terminó la compleción del programa agregando el procesamiento de entrada por consola, y salida en el archivo pasado por parámetro al ejecutar el programa. Este programa sería luego levantado desde la emulación de la máquina virtual con NetBSD que nos permitiría obtener el código assembly de nuestra aplicación, incluido al final de este informe, con el comando de makefile `make asm`.

## 1.2. Implementación

### 1.2.1. void print\_version()

Imprime la version del programa por consola.

### 1.2.2. void print\_usage()

Imprime la leyenda de ayuda del programa por consola.

### 1.2.3. int is\_pal(char\* word)

Recibe un puntero a una cadena de caracteres y devuelve 1 si la misma es un palindromo y 0 si no lo es.

### 1.2.4. void chomp(char\* word)

Recibe un puntero a una cadena de caracteres y elimina el ultimo caracter de la misma si se trata de un salto de linea (el nombre de la funcion fue tomado de la funcion homonima de PERL).

### 1.2.5. void handle\_line(char\* line, FILE\* out\_f)

Recibe un puntero a una linea del archivo y un puntero a un archivo de salida, se encarga de imprimir todos los palindromos encontrados en la linea, en el archivo recibido

### 1.2.6. void handle(char\* input\_file\_name, char\* output\_file\_name)

Recibe punteros a los nombres completos de los archivos de entrada y salida, procede a ejecutar la logica completa del programa (si algun nombre es equivalente a "se reemplaza por la entrada/salida estandar)

### 1.2.7. int main(int argc, char\*\* argv)

Punto de entrada al programa, se procesan los parametros recibidos de linea de comando y se ejecuta la logica del programa de ser correctos, de lo contrario se muestra la leyenda de ayuda.

## 1.3. Compilación

Se ha incluido un archivo Makefile para simplificar la obtención del ejecutable, el mismo puede obtenerse simplemente mediante la ejecución del comando *make*, que generará un archivo binario *tp0*:

```
$ make
gcc -g -Wall -o tp0 tp0.c
$ ls
tp0.c Makefile tp0
```

## 1.4. Ejecución

Una vez obtenido el ejecutable, el mismo se puede ejecutar con el parámetro `-h` para obtener la leyenda de ayuda:

```
$ ./tp0 -h
Usage:
    tp0 -h
    tp0 -V
    tp0 [options]

Options:
    -V, --version    Print version and quit.
    -h, --help       Print this information.
    -i, --input      Location of the input file.
    -o, --output     Location of the output file.

Examples:
    tp0 -i ~/input -o ~/output

Usage:
    tp0 -h
    tp0 -V
    tp0 [options]

Options:
    -V, --version    Print version and quit.
    -h, --help       Print this information.
    -i, --input      Location of the input file.
    -o, --output     Location of the output file.

Examples:
    tp0 -i ~/input -o ~/output
```

O utilizando cualquiera de los parámetros requeridos por el enunciado:

```
$ echo "somos_todos_bob" | ./tp0 -o pal.txt
$ cat pal.txt
somos
bob
```

## 2. Casos de prueba

### 2.1. Makefile

Se puede utilizar el Makefile para correr casos de prueba:

```
$ make test
gcc -g -Wall -o tp0 tp0.c
./tp0 -i tests/test1.in > tests/test1.res
diff tests/test1.out tests/test1.res
./tp0 -i tests/test2.in > tests/test2.res
diff tests/test2.out tests/test2.res
./tp0 -i tests/test3.in > tests/test3.res
diff tests/test3.out tests/test3.res
./tp0 -i tests/test4.in > tests/test4.res
diff tests/test4.out tests/test4.res
```

### 2.2. Validación de parámetros

#### 2.2.1. Input

```
$ ./tp0 -i /tmp/noexiste.txt
No se pudo abrir el archivo de entrada: /tmp/noexiste.txt
```

#### 2.2.2. Output

```
$ echo "bob" |./tp0 -o /tmp/
No se pudo abrir el archivo de salida: /tmp/
```

### 2.3. Utilizado entrada y salida standard

```
$ echo "somos_bob_hope" |./tp0
somos
bob
```

### 2.4. Utilizado archivos

```
$ echo "somos_bob_hope" >> test.txt
$ ./tp0 -i test.txt -o pal.txt
$ cat pal.txt
somos
bob
```

## 3. Código fuente

### 3.1. C

```
1  /*
2  * tpo.h
3  *
4  *   Created on: 24 sep. 2017
5  *       Author: Diego
6  */
7
8  #ifndef TP0.H_
9  #define TP0.H_
10
11 #include <stdio.h>
12 #include <stdlib.h>
13 #include <string.h>
14 #include <ctype.h>
15 #include <errno.h>
16
17 /* Valores posibles de ERRNO para chequear error:
18 *
19 * www.virtsync.com/c-error-codes-include-errno
20 *
21 * Valores que puede tomar ERRNO con fopen:
22 * http://man7.org/linux/man-pages/man3/fopen.3.html#ERRORS
23 *
24 * Valores que puede tomar ERRNO con fclose:
25 * http://man7.org/linux/man-pages/man3/fclose.3.html#ERRORS
26 */
27
28 #define SUCCESS 0
29 #define ERROR_INPUT_FILE 1
30 #define ERROR_OUTPUT_FILE 2
31 #define VERSION "1.5"
32 #define MAXLINELENGTH 256
33
34 /*
35 * Imprime la version actual del proyecto
36 * Sin retorno.
37 */
38 void print_version();
39
40 /*
41 * Imprime los parametros aceptados al ejectuar el programa por
42 *   linea de comandos
43 * Sin retorno.
44 */
45 void print_usage();
46
47 /*
48 * Evalua si la palabra pasada por parametro es un palindromo
49 * PARAMETRO: puntero a cadena de chars 'word' a evaluar
50 * RETORNO: int booleano correspondiente a si la palabra es
51 *   palindromo:
52 *               1 para true, 0 para false.
53 */
54 int es_palindromo(char* word);
55
56 /*
```



```

56  * Procesa los archivos de entrada y salida pasados por parametros ,
    o asigna los
57  * standard inout o output si no fueran dados.
58  * PRAMETROS: dos punteros a cadenas de caracteres correspondientes
    a nombres de archivos
59  *           de input y output. De ser NULL, se tomara
    el archivo standard.
60  * Sin retorno.
61  */
62 void handle(char* input_file_name , char* output_file_name);
63
64 /*
65  * Main para ejecutar el programa. Toma los parametros agregados en
    la linea de comando.
66  * RETORNO: codigo int SUCCES definido como 1.
67  */
68 int main(int argc , char** argv);
69
70 #endif /* TP0_H_ */

```

```

1  #include "tp0.h"
2
3  void print_version() {
4      printf("%s\n", VERSION);
5  }
6
7  void print_usage() {
8      printf("Usage:\n");
9      printf("\ttp0 -h\n");
10     printf("\ttp0 -V\n");
11     printf("\ttp0 [options]\n");
12     printf("Options:\n");
13     printf("\t-V, --version\tPrint version and quit.\n");
14     printf("\t-h, --help\tPrint this information.\n");
15     printf("\t-i, --input\tLocation of the input file.\n");
16     printf("\t-o, --output\tLocation of the output file.\n");
17     printf("Examples:\n");
18     printf("\ttp0 -i ~/input -o ~/output\n");
19 }
20
21 int is_pal(char* word){
22     unsigned int i, j;
23     int is_pal = 1;
24     j = strlen(word)-1;
25     for(i=0; i< (strlen(word)/2); i++, j--) {
26         if (tolower(word[i]) != tolower(word[j])) {
27             is_pal = 0;
28             break;
29         }
30     }
31     return is_pal;
32 }
33
34 void chomp(char* s) {
35     if(s[strlen(s) - 1] == '\n') {
36         s[strlen(s) - 1] = '\0';
37     }
38 }
39
40 void handle_line(char* line, FILE* out_f) {
41     char* word = strtok(line, " ");
42     while (word) {
43         chomp(word);
44         if(is_pal(word)) {
45             fputs(word, out_f);
46             fputs("\n", out_f);
47         }
48         word = strtok(NULL, " ");
49     }
50 }
51
52 void handle(char* input_file_name, char* output_file_name) {
53
54     FILE *in_f;
55     FILE *out_f;
56
57     if (input_file_name != NULL && strcmp(input_file_name, "-")
58         != 0) {
59         in_f = fopen(input_file_name, "r");
60         if (in_f == NULL) {

```

```

61             fprintf(stderr, "ERROR:\nNo se pudo abrir el archivo de entrada: %s\n",
62                     input_file_name);
63             perror("Saliendo con error");
64             exit(ERROR.INPUT.FILE);
65     } else {
66         in_f = stdin;
67     }
68
69     if (output_file_name != NULL && strcmp(output_file_name, "-")
70         != 0) {
71         out_f = fopen(output_file_name, "w");
72
73         if (out_f == NULL) {
74             fprintf(stderr, "No se pudo abrir el archivo de salida: %s\n",
75                     output_file_name);
76             perror("Saliendo con error");
77             exit(ERROR.OUTPUT.FILE);
78         }
79     } else {
80         out_f = stdout;
81     }
82     char line[MAXLINELENGTH];
83     while (fgets(line, sizeof(line), in_f)) {
84         handle_line(line, out_f);
85     }
86
87     if(in_f != NULL && in_f != stdin) {
88         fclose(in_f);
89     }
90
91     if(out_f != NULL && out_f != stdout) {
92         fclose(out_f);
93     }
94
95     exit(SUCCESS);
96 }
97
98 int main(int argc, char** argv) {
99
100     if(argc == 1) {
101         handle(NULL, NULL);
102     } else if (argc == 2) {
103         if(strcmp(argv[1], "-h") == 0 || strcmp(argv[1], "
104             --help")==0) {
105             print_usage();
106         }
107
108         if(strcmp(argv[1], "-V") == 0 || strcmp(argv[1], "
109             --version")==0) {
110             print_version();
111         }
112     } else if(argc == 3) {
113         if(strcmp(argv[1], "-i") == 0) {
114             handle(argv[2], NULL);
115         }

```

```

116
117         if(strcmp(argv[1], "-o") == 0) {
118             handle(NULL, argv[2]);
119         }
120
121     } else if(argc == 5) {
122         if(strcmp(argv[1], "-i") == 0 && strcmp(argv[3], "-o") == 0) {
123             handle(argv[2], argv[4]);
124         }
125
126         if(strcmp(argv[1], "-o") == 0 && strcmp(argv[3], "-i") == 0) {
127             handle(argv[4], argv[2]);
128         }
129     }
130
131     print_usage();
132
133     return SUCCESS;
134 }

```

tp0.c

## 3.2. Assembly (MIPS)

El siguiente es un extracto del código assembly generado con los mismos flags de compilación utilizados para generar al ejecutable más *-O0* para evitar cualquier optimización que pueda alterar el resultado (dentro del entorno MIPS emulado mediante el gxemul y la imagen netbsd):

```
1 $ make asm
2 gcc -g -Wall -O0 -S tp0.c
3 Makefile tp0.c      tp0.s
```

```
1      .section .mdebug.abi32
2      .previous
3      .abicalls
4      .file 1 "tp0.c"
5      .section .debug_abbrev,"",@progbits
6 $Ldebug_abbrev0:
7      .section .debug_info,"",@progbits
8 $Ldebug_info0:
9      .section .debug_line,"",@progbits
10 $Ldebug_line0:
11      .text
12 $Ltext0:
13      .file 2 "/usr/include/mips/int_types.h"
14      .file 3 "/usr/include/sys/ansi.h"
15      .file 4 "/usr/include/mips/ansi.h"
16      .file 5 "/usr/include/stdio.h"
17      .file 6 "/usr/include/mips/types.h"
18      .file 7 "/usr/include/sys/types.h"
19      .file 8 "/usr/include/sys/endian.h"
20      .file 9 "/usr/include/pthread_types.h"
21      .file 10 "/usr/include/stdlib.h"
22      .rdata
23      .align 2
24 $LC0:
25      .ascii "%\n\000"
26      .align 2
27 $LC1:
28      .ascii "1.5\000"
29      .text
30      .align 2
31      .globl print_version
32 $LFB29:
33      .loc 1 3 0
34      .ent print_version
35 print_version:
36      .frame $fp,40,$31      # vars= 0, regs= 3/0, args= 16, extra=
37                               8
38      .mask 0xd0000000,-8
39      .fmask 0x00000000,0
40      .set noreorder
41      .cpload $25
42      .set reorder
43      .subu $sp,$sp,40
44      .cpstore 16
45 $LCFI0:
46      sw $31,32($sp)
47 $LCFI1:
48      sw $fp,28($sp)
49 $LCFI2:
```

[illegible]

```

110     .fmask    0x00000000,0
111     .set      noreorder
112     .cpload   $25
113     .set      reorder
114     subu      $sp,$sp,40
115     .cprestore 16
116 $LCFI5:
117     sw        $31,32($sp)
118 $LCFI6:
119     sw        $fp,28($sp)
120 $LCFI7:
121     sw        $28,24($sp)
122 $LCFI8:
123     move      $fp,$sp
124 $LCFI9:
125     .loc 1 8 0
126     la        $4,$LC2
127     la        $25,printf
128     jal       $31,$25
129     .loc 1 9 0
130     la        $4,$LC3
131     la        $25,printf
132     jal       $31,$25
133     .loc 1 10 0
134     la        $4,$LC4
135     la        $25,printf
136     jal       $31,$25
137     .loc 1 11 0
138     la        $4,$LC5
139     la        $25,printf
140     jal       $31,$25
141     .loc 1 12 0
142     la        $4,$LC6
143     la        $25,printf
144     jal       $31,$25
145     .loc 1 13 0
146     la        $4,$LC7
147     la        $25,printf
148     jal       $31,$25
149     .loc 1 14 0
150     la        $4,$LC8
151     la        $25,printf
152     jal       $31,$25
153     .loc 1 15 0
154     la        $4,$LC9
155     la        $25,printf
156     jal       $31,$25
157     .loc 1 16 0
158     la        $4,$LC10
159     la        $25,printf
160     jal       $31,$25
161     .loc 1 17 0
162     la        $4,$LC11
163     la        $25,printf
164     jal       $31,$25
165     .loc 1 18 0
166     la        $4,$LC12
167     la        $25,printf
168     jal       $31,$25
169     .loc 1 19 0
170     move      $sp,$fp
171     lw        $31,32($sp)

```

```

172     lw    $fp,28($sp)
173     addu   $sp,$sp,40
174     j      $31
175     .end    print_usage
176 $LFE31:
177     .size   print_usage,.-print_usage
178     .align  2
179     .globl  is_pal
180 $LFB33:
181     .loc 1 21 0
182     .ent    is_pal
183 is_pal:
184     .frame  $fp,56,$31      # vars= 16, regs= 3/0, args= 16, extra=
185                          8
186     .mask   0xd0000000,-8
187     .fmask  0x00000000,0
188     .set    noreorder
189     .cload  $25
190     .set    reorder
191     subu    $sp,$sp,56
192     .cprestore 16
193 $LCFI10:
194     sw      $31,48($sp)
195 $LCFI11:
196     sw      $fp,44($sp)
197 $LCFI12:
198     sw      $28,40($sp)
199 $LCFI13:
200     move    $fp,$sp
201 $LCFI14:
202     sw      $4,56($fp)
203     .loc 1 23 0
204 $LBB2:
205     li      $2,1           # 0x1
206     sw      $2,32($fp)
207     .loc 1 24 0
208     lw      $4,56($fp)
209     la      $25,strlen
210     jal     $31,$25
211     addu    $2,$2,-1
212     sw      $2,28($fp)
213     .loc 1 25 0
214     sw      $0,24($fp)
215 $L20:
216     lw      $4,56($fp)
217     la      $25,strlen
218     jal     $31,$25
219     srl     $3,$2,1
220     lw      $2,24($fp)
221     sltu    $2,$2,$3
222     bne     $2,$0,$L23
223     b       $L21
224 $L23:
225     .loc 1 26 0
226     lw      $3,56($fp)
227     lw      $2,24($fp)
228     addu    $2,$3,$2
229     lb      $2,0($2)
230     sll     $3,$2,1
231     lw      $2,_tolower_tab_
232     addu    $2,$3,$2
233     addu    $4,$2,2

```



```

233     lw    $3,56($fp)
234     lw    $2,28($fp)
235     addu   $2,$3,$2
236     lb     $2,0($2)
237     sll    $3,$2,1
238     lw     $2,_tolower_tab_
239     addu   $2,$3,$2
240     addu   $2,$2,2
241     lh     $3,0($4)
242     lh     $2,0($2)
243     beq    $3,$2,$L22
244     .loc 1 27 0
245     sw     $0,32($fp)
246     .loc 1 28 0
247     b      $L21
248     .loc 1 25 0
249 $L22:
250     lw     $2,24($fp)
251     addu   $2,$2,1
252     sw     $2,24($fp)
253     lw     $2,28($fp)
254     addu   $2,$2,-1
255     sw     $2,28($fp)
256     b      $L20
257 $L21:
258     .loc 1 31 0
259     lw     $2,32($fp)
260     .loc 1 32 0
261     move   $sp,$fp
262     lw     $31,48($sp)
263     lw     $fp,44($sp)
264     addu   $sp,$sp,56
265     j      $31
266 $LBE2:
267     .end    is_pal
268 $LFE33:
269     .size   is_pal,.-is_pal
270     .align  2
271     .globl  chomp
272 $LFB35:
273     .loc 1 34 0
274     .ent    chomp
275 chomp:
276     .frame  $fp,40,$31      # vars= 0, regs= 3/0, args= 16, extra=
277     .mask   0xd0000000,-8
278     .fmask  0x00000000,0
279     .set    noreorder
280     .cpld   $25
281     .set    reorder
282     subu    $sp,$sp,40
283     .cprestore 16
284 $LCFI15:
285     sw      $31,32($sp)
286 $LCFI16:
287     sw      $fp,28($sp)
288 $LCFI17:
289     sw      $28,24($sp)
290 $LCFI18:
291     move    $fp,$sp
292 $LCFI19:
293     sw      $4,40($fp)

```

```

294     .loc 1 35 0
295     lw $4,40($fp)
296     la $25,strlen
297     jal $31,$25
298     move $3,$2
299     lw $2,40($fp)
300     addu $2,$3,$2
301     addu $2,$2,-1
302     lb $3,0($2)
303     li $2,10          # 0xa
304     bne $3,$2,$L25
305     .loc 1 36 0
306     lw $4,40($fp)
307     la $25,strlen
308     jal $31,$25
309     move $3,$2
310     lw $2,40($fp)
311     addu $2,$3,$2
312     addu $2,$2,-1
313     sb $0,0($2)
314     .loc 1 38 0
315 $L25:
316     move $sp,$fp
317     lw $31,32($sp)
318     lw $fp,28($sp)
319     addu $sp,$sp,40
320     j $31
321     .end chomp
322 $LFE35:
323     .size chomp,.-chomp
324     .rdata
325     .align 2
326 $LC13:
327     .ascii "\000"
328     .align 2
329 $LC14:
330     .ascii "\n\000"
331     .text
332     .align 2
333     .globl handle_line
334 $LFB37:
335     .loc 1 40 0
336     .ent handle_line
337 handle_line:
338     .frame $fp,48,$31      # vars= 8, regs= 3/0, args= 16, extra=
339                             8
340     .mask 0xd0000000,-8
341     .fmask 0x00000000,0
342     .set noreorder
343     .cpload $25
344     .set reorder
345     subu $sp,$sp,48
346     .cpstore 16
347 $LCFI20:
348     sw $31,40($sp)
349 $LCFI21:
350     sw $fp,36($sp)
351 $LCFI22:
352     sw $28,32($sp)
353 $LCFI23:
354     move $fp,$sp
355 $LCFI24:

```

```

355     sw    $4,48($fp)
356     sw    $5,52($fp)
357     .loc 1 41 0
358 $LBB3:
359     lw    $4,48($fp)
360     la    $5,$LC13
361     la    $25, strtok
362     jal   $31,$25
363     sw    $2,24($fp)
364     .loc 1 42 0
365 $L28:
366     lw    $2,24($fp)
367     bne   $2,$0,$L30
368     b     $L27
369 $L30:
370     .loc 1 43 0
371     lw    $4,24($fp)
372     la    $25, chomp
373     jal   $31,$25
374     .loc 1 44 0
375     lw    $4,24($fp)
376     la    $25, is_pal
377     jal   $31,$25
378     beq   $2,$0,$L31
379     .loc 1 45 0
380     lw    $4,24($fp)
381     lw    $5,52($fp)
382     la    $25, fputs
383     jal   $31,$25
384     .loc 1 46 0
385     la    $4,$LC14
386     lw    $5,52($fp)
387     la    $25, fputs
388     jal   $31,$25
389 $L31:
390     .loc 1 48 0
391     move   $4,$0
392     la    $5,$LC13
393     la    $25, strtok
394     jal   $31,$25
395     sw    $2,24($fp)
396     b     $L28
397     .loc 1 50 0
398 $L27:
399     move   $sp,$fp
400     lw    $31,40($sp)
401     lw    $fp,36($sp)
402     addu   $sp,$sp,48
403     j     $31
404 $LBE3:
405     .end    handle_line
406 $LFE37:
407     .size   handle_line, .-handle_line
408     .rdata
409     .align  2
410 $LC15:
411     .ascii  "-\000"
412     .align  2
413 $LC16:
414     .ascii  "r\000"
415     .align  2
416 $LC17:

```

```

417 .ascii "ERROR:\n"
418 .ascii "No se pudo abrir el archivo de entrada: %s\n\000"
419 .align 2
420 $LC18:
421 .ascii "Saliendo con error\000"
422 .align 2
423 $LC19:
424 .ascii "w\000"
425 .align 2
426 $LC20:
427 .ascii "No se pudo abrir el archivo de salida: %s\n\000"
428 .text
429 .align 2
430 .globl handle
431 $LFB39:
432 .loc 1 52 0
433 .ent handle
434 handle:
435 .frame $fp,304,$31 # vars= 264, regs= 3/0, args= 16, extra
    = 8
436 .mask 0xd0000000,-8
437 .fmask 0x00000000,0
438 .set noreorder
439 .cpload $25
440 .set reorder
441 subu $sp,$sp,304
442 .cpstore 16
443 $LCFI25:
444 sw $31,296($sp)
445 $LCFI26:
446 sw $fp,292($sp)
447 $LCFI27:
448 sw $28,288($sp)
449 $LCFI28:
450 move $fp,$sp
451 $LCFI29:
452 sw $4,304($fp)
453 sw $5,308($fp)
454 .loc 1 57 0
455 $LBB4:
456 lw $2,304($fp)
457 beq $2,$0,$L33
458 lw $4,304($fp)
459 la $5,$LC15
460 la $25,strcmp
461 jal $31,$25
462 beq $2,$0,$L33
463 .loc 1 58 0
464 lw $4,304($fp)
465 la $5,$LC16
466 la $25,fopen
467 jal $31,$25
468 sw $2,24($fp)
469 .loc 1 60 0
470 lw $2,24($fp)
471 bne $2,$0,$L35
472 .loc 1 61 0
473 la $4,--sF+176
474 la $5,$LC17
475 lw $6,304($fp)
476 la $25,fprintf
477 jal $31,$25

```

```

478     .loc 1 62 0
479     la $4,$LC18
480     la $25,perror
481     jal $31,$25
482     .loc 1 63 0
483     li $4,1           # 0x1
484     la $25,exit
485     jal $31,$25
486 $L33:
487     .loc 1 66 0
488     la $2,--sF
489     sw $2,24($fp)
490 $L35:
491     .loc 1 69 0
492     lw $2,308($fp)
493     beq $2,$0,$L36
494     lw $4,308($fp)
495     la $5,$LC15
496     la $25,strcmp
497     jal $31,$25
498     beq $2,$0,$L36
499     .loc 1 70 0
500     lw $4,308($fp)
501     la $5,$LC19
502     la $25,fopen
503     jal $31,$25
504     sw $2,28($fp)
505     .loc 1 72 0
506     lw $2,28($fp)
507     bne $2,$0,$L39
508     .loc 1 73 0
509     la $4,--sF+176
510     la $5,$LC20
511     lw $6,308($fp)
512     la $25,fprintf
513     jal $31,$25
514     .loc 1 74 0
515     la $4,$LC18
516     la $25,perror
517     jal $31,$25
518     .loc 1 75 0
519     li $4,2           # 0x2
520     la $25,exit
521     jal $31,$25
522 $L36:
523     .loc 1 79 0
524     la $2,--sF+88
525     sw $2,28($fp)
526     .loc 1 83 0
527 $L39:
528     addu $2,$fp,32
529     move $4,$2
530     li $5,256         # 0x100
531     lw $6,24($fp)
532     la $25,fgets
533     jal $31,$25
534     bne $2,$0,$L41
535     b $L40
536 $L41:
537     .loc 1 84 0
538     addu $2,$fp,32
539     move $4,$2

```

```

540     lw    $5,28($fp)
541     la    $25,handle_line
542     jal   $31,$25
543     b     $L39
544 $L40:
545     .loc 1 87 0
546     lw    $2,24($fp)
547     beq   $2,$0,$L42
548     lw    $3,24($fp)
549     la    $2,--sF
550     beq   $3,$2,$L42
551     .loc 1 88 0
552     lw    $4,24($fp)
553     la    $25,fclose
554     jal   $31,$25
555 $L42:
556     .loc 1 91 0
557     lw    $2,28($fp)
558     beq   $2,$0,$L43
559     lw    $3,28($fp)
560     la    $2,--sF+88
561     beq   $3,$2,$L43
562     .loc 1 92 0
563     lw    $4,28($fp)
564     la    $25,fclose
565     jal   $31,$25
566 $L43:
567     .loc 1 95 0
568     move   $4,$0
569     la    $25,exit
570     jal   $31,$25
571     .loc 1 96 0
572 $LBE4:
573     .end    handle
574 $LFE39:
575     .size   handle,.-handle
576     .rdata
577     .align  2
578 $LC21:
579     .ascii  "--h\000"
580     .align  2
581 $LC22:
582     .ascii  "--help\000"
583     .align  2
584 $LC23:
585     .ascii  "--V\000"
586     .align  2
587 $LC24:
588     .ascii  "--version\000"
589     .align  2
590 $LC25:
591     .ascii  "--i\000"
592     .align  2
593 $LC26:
594     .ascii  "--o\000"
595     .text
596     .align  2
597     .globl main
598 $LFB41:
599     .loc 1 98 0
600     .ent    main
601 main:

```

```

602     .frame    $fp,40,$31      # vars= 0, regs= 3/0, args= 16, extra=
603         8
604     .mask     0xd0000000,-8
605     .fmask    0x00000000,0
606     .set      noreorder
607     .cpload   $25
608     .set      reorder
609     subu      $sp,$sp,40
610     .cprestore 16
611 $LCFI30:
612     sw        $31,32($sp)
613 $LCFI31:
614     sw        $fp,28($sp)
615 $LCFI32:
616     sw        $28,24($sp)
617 $LCFI33:
618     move      $fp,$sp
619 $LCFI34:
620     sw        $4,40($fp)
621     sw        $5,44($fp)
622     .loc      1 100 0
623     lw        $3,40($fp)
624     li        $2,1            # 0x1
625     bne       $3,$2,$L45
626     .loc      1 101 0
627     move      $4,$0
628     move      $5,$0
629     la        $25,handle
630     jal       $31,$25
631     b         $L46
632 $L45:
633     .loc      1 103 0
634     lw        $3,40($fp)
635     li        $2,2            # 0x2
636     bne       $3,$2,$L47
637     .loc      1 104 0
638     lw        $2,44($fp)
639     addu      $2,$2,4
640     lw        $4,0($2)
641     la        $5,$LC21
642     la        $25,strcmp
643     jal       $31,$25
644     beq       $2,$0,$L49
645     lw        $2,44($fp)
646     addu      $2,$2,4
647     lw        $4,0($2)
648     la        $5,$LC22
649     la        $25,strcmp
650     jal       $31,$25
651     bne       $2,$0,$L48
652 $L49:
653     .loc      1 105 0
654     la        $25,print_usage
655     jal       $31,$25
656 $L48:
657     .loc      1 108 0
658     lw        $2,44($fp)
659     addu      $2,$2,4
660     lw        $4,0($2)
661     la        $5,$LC23
662     la        $25,strcmp
663     jal       $31,$25

```

```

663     beq $2,$0,$L51
664     lw  $2,44($fp)
665     addu $2,$2,4
666     lw  $4,0($2)
667     la  $5,$LC24
668     la  $25,strcmp
669     jal $31,$25
670     bne $2,$0,$L46
671 $L51:
672     .loc 1 109 0
673     la  $25,print_version
674     jal $31,$25
675     b   $L46
676 $L47:
677     .loc 1 112 0
678     lw  $3,40($fp)
679     li  $2,3           # 0x3
680     bne $3,$2,$L53
681     .loc 1 113 0
682     lw  $2,44($fp)
683     addu $2,$2,4
684     lw  $4,0($2)
685     la  $5,$LC25
686     la  $25,strcmp
687     jal $31,$25
688     bne $2,$0,$L54
689     .loc 1 114 0
690     lw  $2,44($fp)
691     addu $2,$2,8
692     lw  $4,0($2)
693     move $5,$0
694     la  $25,handle
695     jal $31,$25
696 $L54:
697     .loc 1 117 0
698     lw  $2,44($fp)
699     addu $2,$2,4
700     lw  $4,0($2)
701     la  $5,$LC26
702     la  $25,strcmp
703     jal $31,$25
704     bne $2,$0,$L46
705     .loc 1 118 0
706     lw  $2,44($fp)
707     addu $2,$2,8
708     move $4,$0
709     lw  $5,0($2)
710     la  $25,handle
711     jal $31,$25
712     b   $L46
713 $L53:
714     .loc 1 121 0
715     lw  $3,40($fp)
716     li  $2,5           # 0x5
717     bne $3,$2,$L46
718     .loc 1 122 0
719     lw  $2,44($fp)
720     addu $2,$2,4
721     lw  $4,0($2)
722     la  $5,$LC25
723     la  $25,strcmp
724     jal $31,$25

```



```

725     bne $2,$0,$L58
726     lw  $2,44($fp)
727     addu $2,$2,12
728     lw  $4,0($2)
729     la  $5,$LC26
730     la  $25,strcmp
731     jal $31,$25
732     bne $2,$0,$L58
733     .loc 1 123 0
734     lw  $2,44($fp)
735     addu $3,$2,8
736     lw  $2,44($fp)
737     addu $2,$2,16
738     lw  $4,0($3)
739     lw  $5,0($2)
740     la  $25,handle
741     jal $31,$25
742 $L58:
743     .loc 1 126 0
744     lw  $2,44($fp)
745     addu $2,$2,4
746     lw  $4,0($2)
747     la  $5,$LC26
748     la  $25,strcmp
749     jal $31,$25
750     bne $2,$0,$L46
751     lw  $2,44($fp)
752     addu $2,$2,12
753     lw  $4,0($2)
754     la  $5,$LC25
755     la  $25,strcmp
756     jal $31,$25
757     bne $2,$0,$L46
758     .loc 1 127 0
759     lw  $2,44($fp)
760     addu $3,$2,16
761     lw  $2,44($fp)
762     addu $2,$2,8
763     lw  $4,0($3)
764     lw  $5,0($2)
765     la  $25,handle
766     jal $31,$25
767 $L46:
768     .loc 1 131 0
769     la  $25,print_usage
770     jal $31,$25
771     .loc 1 133 0
772     move $2,$0
773     .loc 1 134 0
774     move $sp,$fp
775     lw  $31,32($sp)
776     lw  $fp,28($sp)
777     addu $sp,$sp,40
778     j   $31
779     .end    main
780 $LFE41:
781     .size   main,.-main
782     .file 11 "/usr/include/ctype.h"
783     .section .debug_frame,"",@progbits
784 $Lframe0:
785     .4byte $LECIE0-$LSCIE0
786 $LSCIE0:

```

```

787 .4 byte 0xffffffff
788 .byte 0x1
789 .ascii "\000"
790 .uleb128 0x1
791 .sleb128 4
792 .byte 0x40
793 .byte 0xc
794 .uleb128 0x1d
795 .uleb128 0x0
796 .align 2
797 $LECIE0:
798 $LSFDE0:
799 .4 byte $LEFDE0-$LASFDE0
800 $LASFDE0:
801 .4 byte $Lframe0
802 .4 byte $LFB29
803 .4 byte $LFE29-$LFB29
804 .byte 0x4
805 .4 byte $LCFI0-$LFB29
806 .byte 0xe
807 .uleb128 0x28
808 .byte 0x4
809 .4 byte $LCFI3-$LCFI0
810 .byte 0x11
811 .uleb128 0x1c
812 .sleb128 -4
813 .byte 0x11
814 .uleb128 0x1e
815 .sleb128 -3
816 .byte 0x11
817 .uleb128 0x40
818 .sleb128 -2
819 .byte 0x4
820 .4 byte $LCFI4-$LCFI3
821 .byte 0xc
822 .uleb128 0x1e
823 .uleb128 0x28
824 .align 2
825 $LEFDE0:
826 $LSFDE2:
827 .4 byte $LEFDE2-$LASFDE2
828 $LASFDE2:
829 .4 byte $Lframe0
830 .4 byte $LFB31
831 .4 byte $LFE31-$LFB31
832 .byte 0x4
833 .4 byte $LCFI5-$LFB31
834 .byte 0xe
835 .uleb128 0x28
836 .byte 0x4
837 .4 byte $LCFI8-$LCFI5
838 .byte 0x11
839 .uleb128 0x1c
840 .sleb128 -4
841 .byte 0x11
842 .uleb128 0x1e
843 .sleb128 -3
844 .byte 0x11
845 .uleb128 0x40
846 .sleb128 -2
847 .byte 0x4
848 .4 byte $LCFI9-$LCFI8

```

```

849     .byte    0xc
850     .uleb128 0x1e
851     .uleb128 0x28
852     .align   2
853 $LEFDE2:
854 $LSFDE4:
855     .4 byte   $LEFDE4-$LASFDE4
856 $LASFDE4:
857     .4 byte   $Lframe0
858     .4 byte   $LFB33
859     .4 byte   $LFE33-$LFB33
860     .byte     0x4
861     .4 byte   $LCFI10-$LFB33
862     .byte     0xe
863     .uleb128 0x38
864     .byte     0x4
865     .4 byte   $LCFI13-$LCFI10
866     .byte     0x11
867     .uleb128 0x1c
868     .sleb128 -4
869     .byte     0x11
870     .uleb128 0x1e
871     .sleb128 -3
872     .byte     0x11
873     .uleb128 0x40
874     .sleb128 -2
875     .byte     0x4
876     .4 byte   $LCFI14-$LCFI13
877     .byte     0xc
878     .uleb128 0x1e
879     .uleb128 0x38
880     .align   2
881 $LEFDE4:
882 $LSFDE6:
883     .4 byte   $LEFDE6-$LASFDE6
884 $LASFDE6:
885     .4 byte   $Lframe0
886     .4 byte   $LFB35
887     .4 byte   $LFE35-$LFB35
888     .byte     0x4
889     .4 byte   $LCFI15-$LFB35
890     .byte     0xe
891     .uleb128 0x28
892     .byte     0x4
893     .4 byte   $LCFI18-$LCFI15
894     .byte     0x11
895     .uleb128 0x1c
896     .sleb128 -4
897     .byte     0x11
898     .uleb128 0x1e
899     .sleb128 -3
900     .byte     0x11
901     .uleb128 0x40
902     .sleb128 -2
903     .byte     0x4
904     .4 byte   $LCFI19-$LCFI18
905     .byte     0xc
906     .uleb128 0x1e
907     .uleb128 0x28
908     .align   2
909 $LEFDE6:
910 $LSFDE8:

```

```

911     .4 byte    $LEFDE8-$LASFDE8
912 $LASFDE8:
913     .4 byte    $Lframe0
914     .4 byte    $LFB37
915     .4 byte    $LFE37-$LFB37
916     . byte     0x4
917     .4 byte    $LCFI20-$LFB37
918     . byte     0xe
919     .uleb128   0x30
920     . byte     0x4
921     .4 byte    $LCFI23-$LCFI20
922     . byte     0x11
923     .uleb128   0x1c
924     .sleb128   -4
925     . byte     0x11
926     .uleb128   0x1e
927     .sleb128   -3
928     . byte     0x11
929     .uleb128   0x40
930     .sleb128   -2
931     . byte     0x4
932     .4 byte    $LCFI24-$LCFI23
933     . byte     0xc
934     .uleb128   0x1e
935     .uleb128   0x30
936     . align    2
937 $LEFDE8:
938 $LSFDE10:
939     .4 byte    $LEFDE10-$LASFDE10
940 $LASFDE10:
941     .4 byte    $Lframe0
942     .4 byte    $LFB39
943     .4 byte    $LFE39-$LFB39
944     . byte     0x4
945     .4 byte    $LCFI25-$LFB39
946     . byte     0xe
947     .uleb128   0x130
948     . byte     0x4
949     .4 byte    $LCFI28-$LCFI25
950     . byte     0x11
951     .uleb128   0x1c
952     .sleb128   -4
953     . byte     0x11
954     .uleb128   0x1e
955     .sleb128   -3
956     . byte     0x11
957     .uleb128   0x40
958     .sleb128   -2
959     . byte     0x4
960     .4 byte    $LCFI29-$LCFI28
961     . byte     0xc
962     .uleb128   0x1e
963     .uleb128   0x130
964     . align    2
965 $LEFDE10:
966 $LSFDE12:
967     .4 byte    $LEFDE12-$LASFDE12
968 $LASFDE12:
969     .4 byte    $Lframe0
970     .4 byte    $LFB41
971     .4 byte    $LFE41-$LFB41
972     . byte     0x4

```

```

973     .4 byte    $LCFI30-$LFB41
974     .byte     0xe
975     .uleb128   0x28
976     .byte     0x4
977     .4 byte    $LCFI33-$LCFI30
978     .byte     0x11
979     .uleb128   0x1c
980     .sleb128   -4
981     .byte     0x11
982     .uleb128   0x1e
983     .sleb128   -3
984     .byte     0x11
985     .uleb128   0x40
986     .sleb128   -2
987     .byte     0x4
988     .4 byte    $LCFI34-$LCFI33
989     .byte     0xc
990     .uleb128   0x1e
991     .uleb128   0x28
992     .align     2
993 $LEFDE12:
994     .align     0
995     .text
996 $Letext0:
997     .section    .debug_info
998     .4 byte     0xd38
999     .2 byte     0x2
1000    .4 byte     $Ldebug_abbrev0
1001    .byte       0x4
1002    .uleb128     0x1
1003    .4 byte     $Ldebug_line0
1004    .4 byte     $Letext0
1005    .4 byte     $Ltext0
1006    .4 byte     $LC245
1007    .4 byte     $LC246
1008    .4 byte     $LC247
1009    .byte       0x1
1010    .uleb128     0x2
1011    .4 byte     $LC27
1012    .byte       0x2
1013    .byte       0x30
1014    .4 byte     0x30
1015    .uleb128     0x3
1016    .4 byte     $LC29
1017    .byte       0x1
1018    .byte       0x6
1019    .uleb128     0x2
1020    .4 byte     $LC28
1021    .byte       0x2
1022    .byte       0x31
1023    .4 byte     0x42
1024    .uleb128     0x3
1025    .4 byte     $LC30
1026    .byte       0x1
1027    .byte       0x8
1028    .uleb128     0x2
1029    .4 byte     $LC31
1030    .byte       0x2
1031    .byte       0x32
1032    .4 byte     0x54
1033    .uleb128     0x3
1034    .4 byte     $LC32

```

```

1035 .byte 0x2
1036 .byte 0x5
1037 .uleb128 0x2
1038 .4 byte $LC33
1039 .byte 0x2
1040 .byte 0x33
1041 .4 byte 0x66
1042 .uleb128 0x3
1043 .4 byte $LC34
1044 .byte 0x2
1045 .byte 0x7
1046 .uleb128 0x2
1047 .4 byte $LC35
1048 .byte 0x2
1049 .byte 0x34
1050 .4 byte 0x78
1051 .uleb128 0x4
1052 .ascii "int\000"
1053 .byte 0x4
1054 .byte 0x5
1055 .uleb128 0x2
1056 .4 byte $LC36
1057 .byte 0x2
1058 .byte 0x35
1059 .4 byte 0x8a
1060 .uleb128 0x3
1061 .4 byte $LC37
1062 .byte 0x4
1063 .byte 0x7
1064 .uleb128 0x2
1065 .4 byte $LC38
1066 .byte 0x2
1067 .byte 0x3e
1068 .4 byte 0x9c
1069 .uleb128 0x3
1070 .4 byte $LC39
1071 .byte 0x8
1072 .byte 0x5
1073 .uleb128 0x2
1074 .4 byte $LC40
1075 .byte 0x2
1076 .byte 0x40
1077 .4 byte 0xae
1078 .uleb128 0x3
1079 .4 byte $LC41
1080 .byte 0x8
1081 .byte 0x7
1082 .uleb128 0x2
1083 .4 byte $LC42
1084 .byte 0x2
1085 .byte 0x4b
1086 .4 byte 0x78
1087 .uleb128 0x2
1088 .4 byte $LC43
1089 .byte 0x2
1090 .byte 0x4c
1091 .4 byte 0x8a
1092 .uleb128 0x2
1093 .4 byte $LC44
1094 .byte 0x3
1095 .byte 0x2c
1096 .4 byte 0xd6

```

1097	.uleb128 0x5
1098	.byte 0x4
1099	.4 byte 0xdc
1100	.uleb128 0x3
1101	.4 byte \$LC45
1102	.byte 0x1
1103	.byte 0x6
1104	.uleb128 0x2
1105	.4 byte \$LC46
1106	.byte 0x3
1107	.byte 0x2d
1108	.4 byte 0x7f
1109	.uleb128 0x2
1110	.4 byte \$LC47
1111	.byte 0x3
1112	.byte 0x2e
1113	.4 byte 0x7f
1114	.uleb128 0x2
1115	.4 byte \$LC48
1116	.byte 0x3
1117	.byte 0x2f
1118	.4 byte 0x5b
1119	.uleb128 0x2
1120	.4 byte \$LC49
1121	.byte 0x3
1122	.byte 0x30
1123	.4 byte 0x7f
1124	.uleb128 0x2
1125	.4 byte \$LC50
1126	.byte 0x3
1127	.byte 0x31
1128	.4 byte 0x91
1129	.uleb128 0x2
1130	.4 byte \$LC51
1131	.byte 0x3
1132	.byte 0x32
1133	.4 byte 0x6d
1134	.uleb128 0x2
1135	.4 byte \$LC52
1136	.byte 0x3
1137	.byte 0x33
1138	.4 byte 0x37
1139	.uleb128 0x2
1140	.4 byte \$LC53
1141	.byte 0x3
1142	.byte 0x34
1143	.4 byte 0x8a
1144	.uleb128 0x2
1145	.4 byte \$LC54
1146	.byte 0x3
1147	.byte 0x35
1148	.4 byte 0x7f
1149	.uleb128 0x2
1150	.4 byte \$LC55
1151	.byte 0x3
1152	.byte 0x36
1153	.4 byte 0xa3
1154	.uleb128 0x2
1155	.4 byte \$LC56
1156	.byte 0x3
1157	.byte 0x37
1158	.4 byte 0xa3

1159	.uleb128	0x6
1160	.4 byte	0x17b
1161	.byte	0x80
1162	.byte	0x4
1163	.byte	0x65
1164	.uleb128	0x7
1165	.4 byte	\$LC57
1166	.byte	0x4
1167	.byte	0x63
1168	.4 byte	0x17b
1169	.uleb128	0x7
1170	.4 byte	\$LC58
1171	.byte	0x4
1172	.byte	0x64
1173	.4 byte	0x91
1174	.byte	0x0
1175	.uleb128	0x8
1176	.4 byte	0x18b
1177	.4 byte	0xdc
1178	.uleb128	0x9
1179	.4 byte	0x18b
1180	.byte	0x7f
1181	.byte	0x0
1182	.uleb128	0x3
1183	.4 byte	\$LC37
1184	.byte	0x4
1185	.byte	0x7
1186	.uleb128	0x2
1187	.4 byte	\$LC59
1188	.byte	0x4
1189	.byte	0x65
1190	.4 byte	0x15c
1191	.uleb128	0x2
1192	.4 byte	\$LC60
1193	.byte	0x5
1194	.byte	0x2e
1195	.4 byte	0x8a
1196	.uleb128	0x2
1197	.4 byte	\$LC61
1198	.byte	0x5
1199	.byte	0x3a
1200	.4 byte	0x10f
1201	.uleb128	0xa
1202	.4 byte	0x1dc
1203	.4 byte	\$LC64
1204	.byte	0x8
1205	.byte	0x5
1206	.byte	0x4a
1207	.uleb128	0xb
1208	.4 byte	\$LC62
1209	.byte	0x5
1210	.byte	0x4b
1211	.4 byte	0x1dc
1212	.byte	0x2
1213	.byte	0x10
1214	.uleb128	0x0
1215	.uleb128	0xb
1216	.4 byte	\$LC63
1217	.byte	0x5
1218	.byte	0x4c
1219	.4 byte	0x78
1220	.byte	0x2



```

1221 .byte 0x10
1222 .uleb128 0x4
1223 .byte 0x0
1224 .uleb128 0x5
1225 .byte 0x4
1226 .4 byte 0x42
1227 .uleb128 0xa
1228 .4 byte 0x304
1229 .4 byte $LC65
1230 .byte 0x58
1231 .byte 0x5
1232 .byte 0x69
1233 .uleb128 0xc
1234 .ascii "_p\000"
1235 .byte 0x5
1236 .byte 0x6a
1237 .4 byte 0x1dc
1238 .byte 0x2
1239 .byte 0x10
1240 .uleb128 0x0
1241 .uleb128 0xc
1242 .ascii "_r\000"
1243 .byte 0x5
1244 .byte 0x6b
1245 .4 byte 0x78
1246 .byte 0x2
1247 .byte 0x10
1248 .uleb128 0x4
1249 .uleb128 0xc
1250 .ascii "_w\000"
1251 .byte 0x5
1252 .byte 0x6c
1253 .4 byte 0x78
1254 .byte 0x2
1255 .byte 0x10
1256 .uleb128 0x8
1257 .uleb128 0xb
1258 .4 byte $LC66
1259 .byte 0x5
1260 .byte 0x6d
1261 .4 byte 0x54
1262 .byte 0x2
1263 .byte 0x10
1264 .uleb128 0xc
1265 .uleb128 0xb
1266 .4 byte $LC67
1267 .byte 0x5
1268 .byte 0x6e
1269 .4 byte 0x54
1270 .byte 0x2
1271 .byte 0x10
1272 .uleb128 0xe
1273 .uleb128 0xc
1274 .ascii "_bf\000"
1275 .byte 0x5
1276 .byte 0x6f
1277 .4 byte 0x1b3
1278 .byte 0x2
1279 .byte 0x10
1280 .uleb128 0x10
1281 .uleb128 0xb
1282 .4 byte $LC68

```

1283	.byte	0x5
1284	.byte	0x70
1285	.4 byte	0x78
1286	.byte	0x2
1287	.byte	0x10
1288	.uleb128	0x18
1289	.uleb128	0xb
1290	.4 byte	\$LC69
1291	.byte	0x5
1292	.byte	0x73
1293	.4 byte	0x304
1294	.byte	0x2
1295	.byte	0x10
1296	.uleb128	0x1c
1297	.uleb128	0xb
1298	.4 byte	\$LC70
1299	.byte	0x5
1300	.byte	0x74
1301	.4 byte	0x316
1302	.byte	0x2
1303	.byte	0x10
1304	.uleb128	0x20
1305	.uleb128	0xb
1306	.4 byte	\$LC71
1307	.byte	0x5
1308	.byte	0x75
1309	.4 byte	0x336
1310	.byte	0x2
1311	.byte	0x10
1312	.uleb128	0x24
1313	.uleb128	0xb
1314	.4 byte	\$LC72
1315	.byte	0x5
1316	.byte	0x76
1317	.4 byte	0x356
1318	.byte	0x2
1319	.byte	0x10
1320	.uleb128	0x28
1321	.uleb128	0xb
1322	.4 byte	\$LC73
1323	.byte	0x5
1324	.byte	0x77
1325	.4 byte	0x381
1326	.byte	0x2
1327	.byte	0x10
1328	.uleb128	0x2c
1329	.uleb128	0xb
1330	.4 byte	\$LC74
1331	.byte	0x5
1332	.byte	0x7a
1333	.4 byte	0x1b3
1334	.byte	0x2
1335	.byte	0x10
1336	.uleb128	0x30
1337	.uleb128	0xc
1338	.ascii	"_up\000"
1339	.byte	0x5
1340	.byte	0x7d
1341	.4 byte	0x1dc
1342	.byte	0x2
1343	.byte	0x10
1344	.uleb128	0x38

```

1345 .uleb128 0xc
1346 .ascii  "_ur\000"
1347 .byte  0x5
1348 .byte  0x7e
1349 .4 byte 0x78
1350 .byte  0x2
1351 .byte  0x10
1352 .uleb128 0x3c
1353 .uleb128 0xb
1354 .4 byte $LC75
1355 .byte  0x5
1356 .byte  0x81
1357 .4 byte 0x387
1358 .byte  0x2
1359 .byte  0x10
1360 .uleb128 0x40
1361 .uleb128 0xb
1362 .4 byte $LC76
1363 .byte  0x5
1364 .byte  0x82
1365 .4 byte 0x397
1366 .byte  0x2
1367 .byte  0x10
1368 .uleb128 0x43
1369 .uleb128 0xc
1370 .ascii  "_lb\000"
1371 .byte  0x5
1372 .byte  0x85
1373 .4 byte 0x1b3
1374 .byte  0x2
1375 .byte  0x10
1376 .uleb128 0x44
1377 .uleb128 0xb
1378 .4 byte $LC77
1379 .byte  0x5
1380 .byte  0x88
1381 .4 byte 0x78
1382 .byte  0x2
1383 .byte  0x10
1384 .uleb128 0x4c
1385 .uleb128 0xb
1386 .4 byte $LC78
1387 .byte  0x5
1388 .byte  0x89
1389 .4 byte 0x1a8
1390 .byte  0x2
1391 .byte  0x10
1392 .uleb128 0x50
1393 .byte  0x0
1394 .uleb128 0xd
1395 .byte  0x4
1396 .uleb128 0xe
1397 .4 byte 0x316
1398 .byte  0x1
1399 .4 byte 0x78
1400 .uleb128 0xf
1401 .4 byte 0x304
1402 .byte  0x0
1403 .uleb128 0x5
1404 .byte  0x4
1405 .4 byte 0x306
1406 .uleb128 0xe

```

1407	.4 byte	0x336
1408	.byte	0x1
1409	.4 byte	0x78
1410	.uleb128	0xf
1411	.4 byte	0x304
1412	.uleb128	0xf
1413	.4 byte	0xd6
1414	.uleb128	0xf
1415	.4 byte	0x78
1416	.byte	0x0
1417	.uleb128	0x5
1418	.byte	0x4
1419	.4 byte	0x31c
1420	.uleb128	0xe
1421	.4 byte	0x356
1422	.byte	0x1
1423	.4 byte	0x1a8
1424	.uleb128	0xf
1425	.4 byte	0x304
1426	.uleb128	0xf
1427	.4 byte	0x1a8
1428	.uleb128	0xf
1429	.4 byte	0x78
1430	.byte	0x0
1431	.uleb128	0x5
1432	.byte	0x4
1433	.4 byte	0x33c
1434	.uleb128	0xe
1435	.4 byte	0x376
1436	.byte	0x1
1437	.4 byte	0x78
1438	.uleb128	0xf
1439	.4 byte	0x304
1440	.uleb128	0xf
1441	.4 byte	0x376
1442	.uleb128	0xf
1443	.4 byte	0x78
1444	.byte	0x0
1445	.uleb128	0x5
1446	.byte	0x4
1447	.4 byte	0x37c
1448	.uleb128	0x10
1449	.4 byte	0xdc
1450	.uleb128	0x5
1451	.byte	0x4
1452	.4 byte	0x35c
1453	.uleb128	0x8
1454	.4 byte	0x397
1455	.4 byte	0x42
1456	.uleb128	0x9
1457	.4 byte	0x18b
1458	.byte	0x2
1459	.byte	0x0
1460	.uleb128	0x8
1461	.4 byte	0x3a7
1462	.4 byte	0x42
1463	.uleb128	0x9
1464	.4 byte	0x18b
1465	.byte	0x0
1466	.byte	0x0
1467	.uleb128	0x2
1468	.4 byte	\$LC79

1469	.byte	0x5
1470	.byte	0x8a
1471	.4 byte	0x1e2
1472	.uleb128	0x11
1473	.4 byte	\$LC80
1474	.byte	0x5
1475	.2 byte	0x160
1476	.4 byte	0x10f
1477	.uleb128	0x2
1478	.4 byte	\$LC81
1479	.byte	0x6
1480	.byte	0x3b
1481	.4 byte	0x3c9
1482	.uleb128	0x3
1483	.4 byte	\$LC82
1484	.byte	0x4
1485	.byte	0x5
1486	.uleb128	0x2
1487	.4 byte	\$LC83
1488	.byte	0x6
1489	.byte	0x3c
1490	.4 byte	0x3db
1491	.uleb128	0x3
1492	.4 byte	\$LC84
1493	.byte	0x4
1494	.byte	0x7
1495	.uleb128	0x2
1496	.4 byte	\$LC85
1497	.byte	0x6
1498	.byte	0x3d
1499	.4 byte	0x3c9
1500	.uleb128	0x2
1501	.4 byte	\$LC86
1502	.byte	0x6
1503	.byte	0x46
1504	.4 byte	0x3db
1505	.uleb128	0x2
1506	.4 byte	\$LC87
1507	.byte	0x6
1508	.byte	0x47
1509	.4 byte	0x3db
1510	.uleb128	0x2
1511	.4 byte	\$LC88
1512	.byte	0x6
1513	.byte	0x49
1514	.4 byte	0x3db
1515	.uleb128	0x2
1516	.4 byte	\$LC89
1517	.byte	0x6
1518	.byte	0x4a
1519	.4 byte	0x3db
1520	.uleb128	0x2
1521	.4 byte	\$LC90
1522	.byte	0x6
1523	.byte	0x51
1524	.4 byte	0x3c9
1525	.uleb128	0x2
1526	.4 byte	\$LC91
1527	.byte	0x6
1528	.byte	0x5a
1529	.4 byte	0x42f
1530	.uleb128	0x12

1531	.4 byte	0x78
1532	.uleb128	0x2
1533	.4 byte	\$LC92
1534	.byte	0x7
1535	.byte	0x36
1536	.4 byte	0x25
1537	.uleb128	0x2
1538	.4 byte	\$LC93
1539	.byte	0x7
1540	.byte	0x3b
1541	.4 byte	0x37
1542	.uleb128	0x2
1543	.4 byte	\$LC94
1544	.byte	0x7
1545	.byte	0x40
1546	.4 byte	0x49
1547	.uleb128	0x2
1548	.4 byte	\$LC95
1549	.byte	0x7
1550	.byte	0x45
1551	.4 byte	0x5b
1552	.uleb128	0x2
1553	.4 byte	\$LC96
1554	.byte	0x7
1555	.byte	0x4a
1556	.4 byte	0x6d
1557	.uleb128	0x2
1558	.4 byte	\$LC97
1559	.byte	0x7
1560	.byte	0x4f
1561	.4 byte	0x7f
1562	.uleb128	0x2
1563	.4 byte	\$LC98
1564	.byte	0x7
1565	.byte	0x54
1566	.4 byte	0x91
1567	.uleb128	0x2
1568	.4 byte	\$LC99
1569	.byte	0x7
1570	.byte	0x59
1571	.4 byte	0xa3
1572	.uleb128	0x2
1573	.4 byte	\$LC100
1574	.byte	0x7
1575	.byte	0x5d
1576	.4 byte	0x37
1577	.uleb128	0x2
1578	.4 byte	\$LC101
1579	.byte	0x7
1580	.byte	0x5e
1581	.4 byte	0x5b
1582	.uleb128	0x2
1583	.4 byte	\$LC102
1584	.byte	0x7
1585	.byte	0x5f
1586	.4 byte	0x7f
1587	.uleb128	0x2
1588	.4 byte	\$LC103
1589	.byte	0x7
1590	.byte	0x60
1591	.4 byte	0xa3
1592	.uleb128	0x2

1593	.4 byte	\$LC104
1594	. byte	0x8
1595	. byte	0x3a
1596	.4 byte	0xee
1597	.uleb128	0x2
1598	.4 byte	\$LC105
1599	. byte	0x8
1600	. byte	0x3f
1601	.4 byte	0xf9
1602	.uleb128	0x2
1603	.4 byte	\$LC106
1604	. byte	0x7
1605	. byte	0x65
1606	.4 byte	0x42
1607	.uleb128	0x2
1608	.4 byte	\$LC107
1609	. byte	0x7
1610	. byte	0x66
1611	.4 byte	0x66
1612	.uleb128	0x2
1613	.4 byte	\$LC108
1614	. byte	0x7
1615	. byte	0x67
1616	.4 byte	0x8a
1617	.uleb128	0x2
1618	.4 byte	\$LC109
1619	. byte	0x7
1620	. byte	0x68
1621	.4 byte	0x3db
1622	.uleb128	0x2
1623	.4 byte	\$LC110
1624	. byte	0x7
1625	. byte	0x6a
1626	.4 byte	0x42
1627	.uleb128	0x2
1628	.4 byte	\$LC111
1629	. byte	0x7
1630	. byte	0x6b
1631	.4 byte	0x66
1632	.uleb128	0x2
1633	.4 byte	\$LC112
1634	. byte	0x7
1635	. byte	0x6c
1636	.4 byte	0x8a
1637	.uleb128	0x2
1638	.4 byte	\$LC113
1639	. byte	0x7
1640	. byte	0x6d
1641	.4 byte	0x3db
1642	.uleb128	0x2
1643	.4 byte	\$LC114
1644	. byte	0x7
1645	. byte	0x6f
1646	.4 byte	0x4ef
1647	.uleb128	0x2
1648	.4 byte	\$LC115
1649	. byte	0x7
1650	. byte	0x72
1651	.4 byte	0xa3
1652	.uleb128	0x2
1653	.4 byte	\$LC116
1654	. byte	0x7

1655	.byte	0x73
1656	.4 byte	0x91
1657	.uleb128	0x2
1658	.4 byte	\$LC117
1659	.byte	0x7
1660	.byte	0x74
1661	.4 byte	0x552
1662	.uleb128	0x5
1663	.byte	0x4
1664	.4 byte	0x53c
1665	.uleb128	0x2
1666	.4 byte	\$LC118
1667	.byte	0x7
1668	.byte	0x80
1669	.4 byte	0x91
1670	.uleb128	0x2
1671	.4 byte	\$LC119
1672	.byte	0x7
1673	.byte	0x81
1674	.4 byte	0xa3
1675	.uleb128	0x2
1676	.4 byte	\$LC120
1677	.byte	0x7
1678	.byte	0x83
1679	.4 byte	0x91
1680	.uleb128	0x2
1681	.4 byte	\$LC121
1682	.byte	0x7
1683	.byte	0x84
1684	.4 byte	0x7f
1685	.uleb128	0x2
1686	.4 byte	\$LC122
1687	.byte	0x7
1688	.byte	0x87
1689	.4 byte	0x146
1690	.uleb128	0x2
1691	.4 byte	\$LC123
1692	.byte	0x7
1693	.byte	0x8c
1694	.4 byte	0x151
1695	.uleb128	0x2
1696	.4 byte	\$LC124
1697	.byte	0x7
1698	.byte	0x91
1699	.4 byte	0xcb
1700	.uleb128	0x2
1701	.4 byte	\$LC125
1702	.byte	0x7
1703	.byte	0x99
1704	.4 byte	0x91
1705	.uleb128	0x2
1706	.4 byte	\$LC126
1707	.byte	0x7
1708	.byte	0x9c
1709	.4 byte	0x7f
1710	.uleb128	0x2
1711	.4 byte	\$LC127
1712	.byte	0x7
1713	.byte	0x9d
1714	.4 byte	0x7f
1715	.uleb128	0x2
1716	.4 byte	\$LC128



1717	.byte	0x7
1718	.byte	0xa0
1719	.4 byte	0xe3
1720	.uleb128	0x2
1721	.4 byte	\$LC129
1722	.byte	0x7
1723	.byte	0xa4
1724	.4 byte	0x7f
1725	.uleb128	0x2
1726	.4 byte	\$LC130
1727	.byte	0x7
1728	.byte	0xa5
1729	.4 byte	0x7f
1730	.uleb128	0x2
1731	.4 byte	\$LC131
1732	.byte	0x7
1733	.byte	0xa6
1734	.4 byte	0x3c9
1735	.uleb128	0x2
1736	.4 byte	\$LC132
1737	.byte	0x7
1738	.byte	0xa9
1739	.4 byte	0x104
1740	.uleb128	0x2
1741	.4 byte	\$LC133
1742	.byte	0x7
1743	.byte	0xad
1744	.4 byte	0x7f
1745	.uleb128	0x2
1746	.4 byte	\$LC134
1747	.byte	0x7
1748	.byte	0xb5
1749	.4 byte	0x11a
1750	.uleb128	0x2
1751	.4 byte	\$LC135
1752	.byte	0x7
1753	.byte	0xb8
1754	.4 byte	0x6d
1755	.uleb128	0x2
1756	.4 byte	\$LC136
1757	.byte	0x7
1758	.byte	0xb9
1759	.4 byte	0x53c
1760	.uleb128	0x2
1761	.4 byte	\$LC137
1762	.byte	0x7
1763	.byte	0xba
1764	.4 byte	0x6d
1765	.uleb128	0x2
1766	.4 byte	\$LC138
1767	.byte	0x7
1768	.byte	0xbb
1769	.4 byte	0x6d
1770	.uleb128	0x2
1771	.4 byte	\$LC139
1772	.byte	0x7
1773	.byte	0xbe
1774	.4 byte	0x13b
1775	.uleb128	0x2
1776	.4 byte	\$LC140
1777	.byte	0x7
1778	.byte	0xc2

1779	.4 byte	0x6d
1780	.uleb128	0x2
1781	.4 byte	\$LC141
1782	.byte	0x7
1783	.byte	0xfc
1784	.4 byte	0x3db
1785	.uleb128	0x11
1786	.4 byte	\$LC142
1787	.byte	0x7
1788	.2 byte	0x107
1789	.4 byte	0x78
1790	.uleb128	0x11
1791	.4 byte	\$LC143
1792	.byte	0x7
1793	.2 byte	0x10c
1794	.4 byte	0x3c9
1795	.uleb128	0x11
1796	.4 byte	\$LC144
1797	.byte	0x7
1798	.2 byte	0x111
1799	.4 byte	0x78
1800	.uleb128	0x11
1801	.4 byte	\$LC145
1802	.byte	0x7
1803	.2 byte	0x116
1804	.4 byte	0x78
1805	.uleb128	0x11
1806	.4 byte	\$LC146
1807	.byte	0x7
1808	.2 byte	0x11b
1809	.4 byte	0x78
1810	.uleb128	0x11
1811	.4 byte	\$LC147
1812	.byte	0x7
1813	.2 byte	0x120
1814	.4 byte	0x8a
1815	.uleb128	0x11
1816	.4 byte	\$LC148
1817	.byte	0x7
1818	.2 byte	0x12d
1819	.4 byte	0x6d
1820	.uleb128	0x13
1821	.4 byte	0x6d1
1822	.4 byte	\$LC149
1823	.byte	0x20
1824	.byte	0x7
1825	.2 byte	0x142
1826	.uleb128	0x14
1827	.4 byte	\$LC150
1828	.byte	0x7
1829	.2 byte	0x143
1830	.4 byte	0x6d1
1831	.byte	0x2
1832	.byte	0x10
1833	.uleb128	0x0
1834	.byte	0x0
1835	.uleb128	0x8
1836	.4 byte	0x6e1
1837	.4 byte	0x6a8
1838	.uleb128	0x9
1839	.4 byte	0x18b
1840	.byte	0x7

1841	. byte	0x0
1842	.uleb128	0x11
1843	.4 byte	\$LC149
1844	. byte	0x7
1845	.2 byte	0x144
1846	.4 byte	0x6b4
1847	.uleb128	0x2
1848	.4 byte	\$LC151
1849	. byte	0x9
1850	. byte	0x2e
1851	.4 byte	0x42f
1852	.uleb128	0xa
1853	.4 byte	0x721
1854	.4 byte	\$LC152
1855	. byte	0x8
1856	. byte	0x9
1857	. byte	0x39
1858	.uleb128	0xb
1859	.4 byte	\$LC153
1860	. byte	0x9
1861	. byte	0x39
1862	.4 byte	0x727
1863	. byte	0x2
1864	. byte	0x10
1865	.uleb128	0x0
1866	.uleb128	0xb
1867	.4 byte	\$LC154
1868	. byte	0x9
1869	. byte	0x39
1870	.4 byte	0x72d
1871	. byte	0x2
1872	. byte	0x10
1873	.uleb128	0x4
1874	. byte	0x0
1875	.uleb128	0x15
1876	.4 byte	\$LC242
1877	. byte	0x1
1878	.uleb128	0x5
1879	. byte	0x4
1880	.4 byte	0x721
1881	.uleb128	0x5
1882	. byte	0x4
1883	.4 byte	0x727
1884	.uleb128	0x2
1885	.4 byte	\$LC155
1886	. byte	0x9
1887	. byte	0x47
1888	.4 byte	0x727
1889	.uleb128	0x2
1890	.4 byte	\$LC156
1891	. byte	0x9
1892	. byte	0x48
1893	.4 byte	0x749
1894	.uleb128	0xa
1895	.4 byte	0x780
1896	.4 byte	\$LC157
1897	. byte	0xc
1898	. byte	0x9
1899	. byte	0x3c
1900	.uleb128	0xb
1901	.4 byte	\$LC158
1902	. byte	0x9

1903	.byte	0x56
1904	.4 byte	0x8a
1905	.byte	0x2
1906	.byte	0x10
1907	.uleb128	0x0
1908	.uleb128	0xb
1909	.4 byte	\$LC159
1910	.byte	0x9
1911	.byte	0x58
1912	.4 byte	0x78
1913	.byte	0x2
1914	.byte	0x10
1915	.uleb128	0x4
1916	.uleb128	0xb
1917	.4 byte	\$LC160
1918	.byte	0x9
1919	.byte	0x59
1920	.4 byte	0x304
1921	.byte	0x2
1922	.byte	0x10
1923	.uleb128	0x8
1924	.byte	0x0
1925	.uleb128	0x2
1926	.4 byte	\$LC161
1927	.byte	0x9
1928	.byte	0x49
1929	.4 byte	0x78b
1930	.uleb128	0xa
1931	.4 byte	0x7ec
1932	.4 byte	\$LC162
1933	.byte	0x1c
1934	.byte	0x9
1935	.byte	0x3d
1936	.uleb128	0xb
1937	.4 byte	\$LC163
1938	.byte	0x9
1939	.byte	0x5d
1940	.4 byte	0x8a
1941	.byte	0x2
1942	.byte	0x10
1943	.uleb128	0x0
1944	.uleb128	0xb
1945	.4 byte	\$LC164
1946	.byte	0x9
1947	.byte	0x66
1948	.4 byte	0x42f
1949	.byte	0x2
1950	.byte	0x10
1951	.uleb128	0x4
1952	.uleb128	0xb
1953	.4 byte	\$LC165
1954	.byte	0x9
1955	.byte	0x69
1956	.4 byte	0x42f
1957	.byte	0x2
1958	.byte	0x10
1959	.uleb128	0x8
1960	.uleb128	0xb
1961	.4 byte	\$LC166
1962	.byte	0x9
1963	.byte	0x6a
1964	.4 byte	0x733

1965	.byte	0x2
1966	.byte	0x10
1967	.uleb128	0xc
1968	.uleb128	0xb
1969	.4 byte	\$LC167
1970	.byte	0x9
1971	.byte	0x6b
1972	.4 byte	0x6f8
1973	.byte	0x2
1974	.byte	0x10
1975	.uleb128	0x10
1976	.uleb128	0xb
1977	.4 byte	\$LC168
1978	.byte	0x9
1979	.byte	0x6c
1980	.4 byte	0x304
1981	.byte	0x2
1982	.byte	0x10
1983	.uleb128	0x18
1984	.byte	0x0
1985	.uleb128	0x2
1986	.4 byte	\$LC169
1987	.byte	0x9
1988	.byte	0x4a
1989	.4 byte	0x7f7
1990	.uleb128	0xa
1991	.4 byte	0x820
1992	.4 byte	\$LC170
1993	.byte	0x8
1994	.byte	0x9
1995	.byte	0x3e
1996	.uleb128	0xb
1997	.4 byte	\$LC171
1998	.byte	0x9
1999	.byte	0x7c
2000	.4 byte	0x8a
2001	.byte	0x2
2002	.byte	0x10
2003	.uleb128	0x0
2004	.uleb128	0xb
2005	.4 byte	\$LC172
2006	.byte	0x9
2007	.byte	0x7d
2008	.4 byte	0x304
2009	.byte	0x2
2010	.byte	0x10
2011	.uleb128	0x4
2012	.byte	0x0
2013	.uleb128	0x2
2014	.4 byte	\$LC173
2015	.byte	0x9
2016	.byte	0x4b
2017	.4 byte	0x82b
2018	.uleb128	0xa
2019	.4 byte	0x87e
2020	.4 byte	\$LC174
2021	.byte	0x18
2022	.byte	0x9
2023	.byte	0x3f
2024	.uleb128	0xb
2025	.4 byte	\$LC175
2026	.byte	0x9

2027	. byte	0x85
2028	.4 byte	0x8a
2029	. byte	0x2
2030	. byte	0x10
2031	.uleb128	0x0
2032	.uleb128	0xb
2033	.4 byte	\$LC176
2034	. byte	0x9
2035	. byte	0x88
2036	.4 byte	0x42f
2037	. byte	0x2
2038	. byte	0x10
2039	.uleb128	0x4
2040	.uleb128	0xb
2041	.4 byte	\$LC177
2042	. byte	0x9
2043	. byte	0x89
2044	.4 byte	0x6f8
2045	. byte	0x2
2046	. byte	0x10
2047	.uleb128	0x8
2048	.uleb128	0xb
2049	.4 byte	\$LC178
2050	. byte	0x9
2051	. byte	0x8b
2052	.4 byte	0xa8f
2053	. byte	0x2
2054	. byte	0x10
2055	.uleb128	0x10
2056	.uleb128	0xb
2057	.4 byte	\$LC179
2058	. byte	0x9
2059	. byte	0x8c
2060	.4 byte	0x304
2061	. byte	0x2
2062	. byte	0x10
2063	.uleb128	0x14
2064	. byte	0x0
2065	.uleb128	0x2
2066	.4 byte	\$LC180
2067	. byte	0x9
2068	. byte	0x4c
2069	.4 byte	0x889
2070	.uleb128	0xa
2071	.4 byte	0x8b2
2072	.4 byte	\$LC181
2073	. byte	0x8
2074	. byte	0x9
2075	. byte	0x40
2076	.uleb128	0xb
2077	.4 byte	\$LC182
2078	. byte	0x9
2079	. byte	0x9a
2080	.4 byte	0x8a
2081	. byte	0x2
2082	. byte	0x10
2083	.uleb128	0x0
2084	.uleb128	0xb
2085	.4 byte	\$LC183
2086	. byte	0x9
2087	. byte	0x9b
2088	.4 byte	0x304

2089	.byte	0x2
2090	.byte	0x10
2091	.uleb128	0x4
2092	.byte	0x0
2093	.uleb128	0x2
2094	.4 byte	\$LC184
2095	.byte	0x9
2096	.byte	0x4d
2097	.4 byte	0x8bd
2098	.uleb128	0xa
2099	.4 byte	0x8e6
2100	.4 byte	\$LC185
2101	.byte	0x20
2102	.byte	0x9
2103	.byte	0x4d
2104	.uleb128	0xb
2105	.4 byte	\$LC186
2106	.byte	0x9
2107	.byte	0xa2
2108	.4 byte	0x780
2109	.byte	0x2
2110	.byte	0x10
2111	.uleb128	0x0
2112	.uleb128	0xb
2113	.4 byte	\$LC187
2114	.byte	0x9
2115	.byte	0xa3
2116	.4 byte	0x78
2117	.byte	0x2
2118	.byte	0x10
2119	.uleb128	0x1c
2120	.byte	0x0
2121	.uleb128	0x2
2122	.4 byte	\$LC188
2123	.byte	0x9
2124	.byte	0x4e
2125	.4 byte	0x8f1
2126	.uleb128	0xa
2127	.4 byte	0x928
2128	.4 byte	\$LC189
2129	.byte	0xc
2130	.byte	0x9
2131	.byte	0x4e
2132	.uleb128	0xb
2133	.4 byte	\$LC190
2134	.byte	0x9
2135	.byte	0xa9
2136	.4 byte	0x8a
2137	.byte	0x2
2138	.byte	0x10
2139	.uleb128	0x0
2140	.uleb128	0xb
2141	.4 byte	\$LC191
2142	.byte	0x9
2143	.byte	0xaa
2144	.4 byte	0x42f
2145	.byte	0x2
2146	.byte	0x10
2147	.uleb128	0x4
2148	.uleb128	0xb
2149	.4 byte	\$LC192
2150	.byte	0x9

2151	. byte	0xab
2152	. 4 byte	0x78
2153	. byte	0x2
2154	. byte	0x10
2155	. uleb128	0x8
2156	. byte	0x0
2157	. uleb128	0x2
2158	. 4 byte	\$LC193
2159	. byte	0x9
2160	. byte	0x4f
2161	. 4 byte	0x933
2162	. uleb128	0xa
2163	. 4 byte	0x9a2
2164	. 4 byte	\$LC194
2165	. byte	0x24
2166	. byte	0x9
2167	. byte	0x42
2168	. uleb128	0xb
2169	. 4 byte	\$LC195
2170	. byte	0x9
2171	. byte	0xb9
2172	. 4 byte	0x8a
2173	. byte	0x2
2174	. byte	0x10
2175	. uleb128	0x0
2176	. uleb128	0xb
2177	. 4 byte	\$LC196
2178	. byte	0x9
2179	. byte	0xbc
2180	. 4 byte	0x42f
2181	. byte	0x2
2182	. byte	0x10
2183	. uleb128	0x4
2184	. uleb128	0xb
2185	. 4 byte	\$LC197
2186	. byte	0x9
2187	. byte	0xbe
2188	. 4 byte	0x6f8
2189	. byte	0x2
2190	. byte	0x10
2191	. uleb128	0x8
2192	. uleb128	0xb
2193	. 4 byte	\$LC198
2194	. byte	0x9
2195	. byte	0xbf
2196	. 4 byte	0x6f8
2197	. byte	0x2
2198	. byte	0x10
2199	. uleb128	0x10
2200	. uleb128	0xb
2201	. 4 byte	\$LC199
2202	. byte	0x9
2203	. byte	0xc0
2204	. 4 byte	0x8a
2205	. byte	0x2
2206	. byte	0x10
2207	. uleb128	0x18
2208	. uleb128	0xb
2209	. 4 byte	\$LC200
2210	. byte	0x9
2211	. byte	0xc1
2212	. 4 byte	0x733



2213	.byte	0x2
2214	.byte	0x10
2215	.uleb128	0x1c
2216	.uleb128	0xb
2217	.4 byte	\$LC201
2218	.byte	0x9
2219	.byte	0xc2
2220	.4 byte	0x304
2221	.byte	0x2
2222	.byte	0x10
2223	.uleb128	0x20
2224	.byte	0x0
2225	.uleb128	0x2
2226	.4 byte	\$LC202
2227	.byte	0x9
2228	.byte	0x50
2229	.4 byte	0x9ad
2230	.uleb128	0xa
2231	.4 byte	0x9d6
2232	.4 byte	\$LC203
2233	.byte	0x8
2234	.byte	0x9
2235	.byte	0x43
2236	.uleb128	0xb
2237	.4 byte	\$LC204
2238	.byte	0x9
2239	.byte	0xd2
2240	.4 byte	0x8a
2241	.byte	0x2
2242	.byte	0x10
2243	.uleb128	0x0
2244	.uleb128	0xb
2245	.4 byte	\$LC205
2246	.byte	0x9
2247	.byte	0xd3
2248	.4 byte	0x304
2249	.byte	0x2
2250	.byte	0x10
2251	.uleb128	0x4
2252	.byte	0x0
2253	.uleb128	0x2
2254	.4 byte	\$LC206
2255	.byte	0x9
2256	.byte	0x51
2257	.4 byte	0x9e1
2258	.uleb128	0xa
2259	.4 byte	0xa50
2260	.4 byte	\$LC207
2261	.byte	0x20
2262	.byte	0x9
2263	.byte	0x44
2264	.uleb128	0xb
2265	.4 byte	\$LC208
2266	.byte	0x9
2267	.byte	0xda
2268	.4 byte	0x8a
2269	.byte	0x2
2270	.byte	0x10
2271	.uleb128	0x0
2272	.uleb128	0xb
2273	.4 byte	\$LC209
2274	.byte	0x9

2275	.byte	0xdd
2276	.4 byte	0x42f
2277	.byte	0x2
2278	.byte	0x10
2279	.uleb128	0x4
2280	.uleb128	0xb
2281	.4 byte	\$LC210
2282	.byte	0x9
2283	.byte	0xdf
2284	.4 byte	0x6f8
2285	.byte	0x2
2286	.byte	0x10
2287	.uleb128	0x8
2288	.uleb128	0xb
2289	.4 byte	\$LC211
2290	.byte	0x9
2291	.byte	0xe0
2292	.4 byte	0x8a
2293	.byte	0x2
2294	.byte	0x10
2295	.uleb128	0x10
2296	.uleb128	0xb
2297	.4 byte	\$LC212
2298	.byte	0x9
2299	.byte	0xe1
2300	.4 byte	0x8a
2301	.byte	0x2
2302	.byte	0x10
2303	.uleb128	0x14
2304	.uleb128	0xb
2305	.4 byte	\$LC213
2306	.byte	0x9
2307	.byte	0xe2
2308	.4 byte	0x8a
2309	.byte	0x2
2310	.byte	0x10
2311	.uleb128	0x18
2312	.uleb128	0xb
2313	.4 byte	\$LC214
2314	.byte	0x9
2315	.byte	0xe4
2316	.4 byte	0x304
2317	.byte	0x2
2318	.byte	0x10
2319	.uleb128	0x1c
2320	.byte	0x0
2321	.uleb128	0x2
2322	.4 byte	\$LC215
2323	.byte	0x9
2324	.byte	0x52
2325	.4 byte	0xa5b
2326	.uleb128	0xa
2327	.4 byte	0xa84
2328	.4 byte	\$LC216
2329	.byte	0x8
2330	.byte	0x9
2331	.byte	0x45
2332	.uleb128	0xb
2333	.4 byte	\$LC217
2334	.byte	0x9
2335	.byte	0xeb
2336	.4 byte	0x8a

```

2337 .byte 0x2
2338 .byte 0x10
2339 .uleb128 0x0
2340 .uleb128 0xb
2341 .4 byte $LC218
2342 .byte 0x9
2343 .byte 0xec
2344 .4 byte 0x304
2345 .byte 0x2
2346 .byte 0x10
2347 .uleb128 0x4
2348 .byte 0x0
2349 .uleb128 0x2
2350 .4 byte $LC219
2351 .byte 0x9
2352 .byte 0x53
2353 .4 byte 0x78
2354 .uleb128 0x5
2355 .byte 0x4
2356 .4 byte 0x780
2357 .uleb128 0x2
2358 .4 byte $LC220
2359 .byte 0xa
2360 .byte 0x34
2361 .4 byte 0x78
2362 .uleb128 0x16
2363 .4 byte 0xac5
2364 .byte 0x8
2365 .byte 0xa
2366 .byte 0x3b
2367 .uleb128 0xb
2368 .4 byte $LC221
2369 .byte 0xa
2370 .byte 0x39
2371 .4 byte 0x78
2372 .byte 0x2
2373 .byte 0x10
2374 .uleb128 0x0
2375 .uleb128 0xc
2376 .ascii "rem\000"
2377 .byte 0xa
2378 .byte 0x3a
2379 .4 byte 0x78
2380 .byte 0x2
2381 .byte 0x10
2382 .uleb128 0x4
2383 .byte 0x0
2384 .uleb128 0x2
2385 .4 byte $LC222
2386 .byte 0xa
2387 .byte 0x3b
2388 .4 byte 0xaa0
2389 .uleb128 0x16
2390 .4 byte 0xaf5
2391 .byte 0x8
2392 .byte 0xa
2393 .byte 0x40
2394 .uleb128 0xb
2395 .4 byte $LC221
2396 .byte 0xa
2397 .byte 0x3e
2398 .4 byte 0x3c9

```

```

2399 .byte 0x2
2400 .byte 0x10
2401 .uleb128 0x0
2402 .uleb128 0xc
2403 .ascii "rem\000"
2404 .byte 0xa
2405 .byte 0x3f
2406 .4 byte 0x3c9
2407 .byte 0x2
2408 .byte 0x10
2409 .uleb128 0x4
2410 .byte 0x0
2411 .uleb128 0x2
2412 .4 byte $LC223
2413 .byte 0xa
2414 .byte 0x40
2415 .4 byte 0xad0
2416 .uleb128 0x16
2417 .4 byte 0xb25
2418 .byte 0x10
2419 .byte 0xa
2420 .byte 0x4a
2421 .uleb128 0xb
2422 .4 byte $LC221
2423 .byte 0xa
2424 .byte 0x47
2425 .4 byte 0x9c
2426 .byte 0x2
2427 .byte 0x10
2428 .uleb128 0x0
2429 .uleb128 0xc
2430 .ascii "rem\000"
2431 .byte 0xa
2432 .byte 0x49
2433 .4 byte 0x9c
2434 .byte 0x2
2435 .byte 0x10
2436 .uleb128 0x8
2437 .byte 0x0
2438 .uleb128 0x2
2439 .4 byte $LC224
2440 .byte 0xa
2441 .byte 0x4a
2442 .4 byte 0xb00
2443 .uleb128 0x16
2444 .4 byte 0xb55
2445 .byte 0x10
2446 .byte 0xa
2447 .byte 0x51
2448 .uleb128 0xb
2449 .4 byte $LC221
2450 .byte 0xa
2451 .byte 0x4f
2452 .4 byte 0x53c
2453 .byte 0x2
2454 .byte 0x10
2455 .uleb128 0x0
2456 .uleb128 0xc
2457 .ascii "rem\000"
2458 .byte 0xa
2459 .byte 0x50
2460 .4 byte 0x53c

```

2461	.byte	0x2
2462	.byte	0x10
2463	.uleb128	0x8
2464	.byte	0x0
2465	.uleb128	0x2
2466	.4 byte	\$LC225
2467	.byte	0xa
2468	.byte	0x51
2469	.4 byte	0xb30
2470	.uleb128	0x17
2471	.byte	0x1
2472	.4 byte	\$LC226
2473	.byte	0x1
2474	.byte	0x3
2475	.4 byte	\$LFB29
2476	.4 byte	\$LFE29
2477	.4 byte	\$LSFDE0
2478	.byte	0x1
2479	.byte	0x6e
2480	.uleb128	0x17
2481	.byte	0x1
2482	.4 byte	\$LC227
2483	.byte	0x1
2484	.byte	0x7
2485	.4 byte	\$LFB31
2486	.4 byte	\$LFE31
2487	.4 byte	\$LSFDE2
2488	.byte	0x1
2489	.byte	0x6e
2490	.uleb128	0x18
2491	.4 byte	0xbe0
2492	.byte	0x1
2493	.4 byte	\$LC228
2494	.byte	0x1
2495	.byte	0x15
2496	.byte	0x1
2497	.4 byte	0x78
2498	.4 byte	\$LFB33
2499	.4 byte	\$LFE33
2500	.4 byte	\$LSFDE4
2501	.byte	0x1
2502	.byte	0x6e
2503	.uleb128	0x19
2504	.4 byte	\$LC230
2505	.byte	0x1
2506	.byte	0x15
2507	.4 byte	0xd6
2508	.byte	0x2
2509	.byte	0x91
2510	.sleb128	56
2511	.uleb128	0x1a
2512	.ascii	"i\000"
2513	.byte	0x1
2514	.byte	0x16
2515	.4 byte	0x8a
2516	.byte	0x2
2517	.byte	0x91
2518	.sleb128	24
2519	.uleb128	0x1a
2520	.ascii	"j\000"
2521	.byte	0x1
2522	.byte	0x16

2523	.4 byte	0x8a
2524	.byte	0x2
2525	.byte	0x91
2526	.sleb128	28
2527	.uleb128	0x1b
2528	.4 byte	\$LC228
2529	.byte	0x1
2530	.byte	0x17
2531	.4 byte	0x78
2532	.byte	0x2
2533	.byte	0x91
2534	.sleb128	32
2535	.byte	0x0
2536	.uleb128	0x1c
2537	.4 byte	0xc08
2538	.byte	0x1
2539	.4 byte	\$LC229
2540	.byte	0x1
2541	.byte	0x22
2542	.byte	0x1
2543	.4 byte	\$LFB35
2544	.4 byte	\$LFE35
2545	.4 byte	\$LSFDE6
2546	.byte	0x1
2547	.byte	0x6e
2548	.uleb128	0x1d
2549	.ascii	"s\000"
2550	.byte	0x1
2551	.byte	0x22
2552	.4 byte	0xd6
2553	.byte	0x2
2554	.byte	0x91
2555	.sleb128	40
2556	.byte	0x0
2557	.uleb128	0x1c
2558	.4 byte	0xc4e
2559	.byte	0x1
2560	.4 byte	\$LC231
2561	.byte	0x1
2562	.byte	0x28
2563	.byte	0x1
2564	.4 byte	\$LFB37
2565	.4 byte	\$LFE37
2566	.4 byte	\$LSFDE8
2567	.byte	0x1
2568	.byte	0x6e
2569	.uleb128	0x19
2570	.4 byte	\$LC232
2571	.byte	0x1
2572	.byte	0x28
2573	.4 byte	0xd6
2574	.byte	0x2
2575	.byte	0x91
2576	.sleb128	48
2577	.uleb128	0x19
2578	.4 byte	\$LC233
2579	.byte	0x1
2580	.byte	0x28
2581	.4 byte	0xc4e
2582	.byte	0x2
2583	.byte	0x91
2584	.sleb128	52

2585	.uleb128	0x1b
2586	.4 byte	\$LC230
2587	.byte	0x1
2588	.byte	0x29
2589	.4 byte	0xd6
2590	.byte	0x2
2591	.byte	0x91
2592	.sleb128	24
2593	.byte	0x0
2594	.uleb128	0x5
2595	.byte	0x4
2596	.4 byte	0x3a7
2597	.uleb128	0x1c
2598	.4 byte	0xcb8
2599	.byte	0x1
2600	.4 byte	\$LC234
2601	.byte	0x1
2602	.byte	0x34
2603	.byte	0x1
2604	.4 byte	\$LFB39
2605	.4 byte	\$LFE39
2606	.4 byte	\$LSFDE10
2607	.byte	0x1
2608	.byte	0x6e
2609	.uleb128	0x19
2610	.4 byte	\$LC235
2611	.byte	0x1
2612	.byte	0x34
2613	.4 byte	0xd6
2614	.byte	0x3
2615	.byte	0x91
2616	.sleb128	304
2617	.uleb128	0x19
2618	.4 byte	\$LC236
2619	.byte	0x1
2620	.byte	0x34
2621	.4 byte	0xd6
2622	.byte	0x3
2623	.byte	0x91
2624	.sleb128	308
2625	.uleb128	0x1b
2626	.4 byte	\$LC237
2627	.byte	0x1
2628	.byte	0x36
2629	.4 byte	0xc4e
2630	.byte	0x2
2631	.byte	0x91
2632	.sleb128	24
2633	.uleb128	0x1b
2634	.4 byte	\$LC233
2635	.byte	0x1
2636	.byte	0x37
2637	.4 byte	0xc4e
2638	.byte	0x2
2639	.byte	0x91
2640	.sleb128	28
2641	.uleb128	0x1b
2642	.4 byte	\$LC232
2643	.byte	0x1
2644	.byte	0x52
2645	.4 byte	0xcb8
2646	.byte	0x2

2647	.byte	0x91
2648	.sleb128	32
2649	.byte	0x0
2650	.uleb128	0x8
2651	.4 byte	0xcc8
2652	.4 byte	0xdc
2653	.uleb128	0x9
2654	.4 byte	0x18b
2655	.byte	0xff
2656	.byte	0x0
2657	.uleb128	0x18
2658	.4 byte	0xd04
2659	.byte	0x1
2660	.4 byte	\$LC238
2661	.byte	0x1
2662	.byte	0x62
2663	.byte	0x1
2664	.4 byte	0x78
2665	.4 byte	\$LFB41
2666	.4 byte	\$LFE41
2667	.4 byte	\$LSFDE12
2668	.byte	0x1
2669	.byte	0x6e
2670	.uleb128	0x19
2671	.4 byte	\$LC239
2672	.byte	0x1
2673	.byte	0x62
2674	.4 byte	0x78
2675	.byte	0x2
2676	.byte	0x91
2677	.sleb128	40
2678	.uleb128	0x19
2679	.4 byte	\$LC240
2680	.byte	0x1
2681	.byte	0x62
2682	.4 byte	0xd04
2683	.byte	0x2
2684	.byte	0x91
2685	.sleb128	44
2686	.byte	0x0
2687	.uleb128	0x5
2688	.byte	0x4
2689	.4 byte	0xd6
2690	.uleb128	0x1e
2691	.byte	0x1
2692	.4 byte	0x3a7
2693	.uleb128	0x1f
2694	.4 byte	\$LC241
2695	.byte	0x5
2696	.byte	0x8d
2697	.4 byte	0xd0a
2698	.byte	0x1
2699	.byte	0x1
2700	.uleb128	0x15
2701	.4 byte	\$LC243
2702	.byte	0x1
2703	.uleb128	0x1f
2704	.4 byte	\$LC244
2705	.byte	0xb
2706	.byte	0x37
2707	.4 byte	0xd30
2708	.byte	0x1



```

2709 .byte 0x1
2710 .uleb128 0x5
2711 .byte 0x4
2712 .4byte 0xd36
2713 .uleb128 0x10
2714 .4byte 0x54
2715 .byte 0x0
2716 .section .debug_abbrev
2717 .uleb128 0x1
2718 .uleb128 0x11
2719 .byte 0x1
2720 .uleb128 0x10
2721 .uleb128 0x6
2722 .uleb128 0x12
2723 .uleb128 0x1
2724 .uleb128 0x11
2725 .uleb128 0x1
2726 .uleb128 0x3
2727 .uleb128 0xe
2728 .uleb128 0x1b
2729 .uleb128 0xe
2730 .uleb128 0x25
2731 .uleb128 0xe
2732 .uleb128 0x13
2733 .uleb128 0xb
2734 .byte 0x0
2735 .byte 0x0
2736 .uleb128 0x2
2737 .uleb128 0x16
2738 .byte 0x0
2739 .uleb128 0x3
2740 .uleb128 0xe
2741 .uleb128 0x3a
2742 .uleb128 0xb
2743 .uleb128 0x3b
2744 .uleb128 0xb
2745 .uleb128 0x49
2746 .uleb128 0x13
2747 .byte 0x0
2748 .byte 0x0
2749 .uleb128 0x3
2750 .uleb128 0x24
2751 .byte 0x0
2752 .uleb128 0x3
2753 .uleb128 0xe
2754 .uleb128 0xb
2755 .uleb128 0xb
2756 .uleb128 0x3e
2757 .uleb128 0xb
2758 .byte 0x0
2759 .byte 0x0
2760 .uleb128 0x4
2761 .uleb128 0x24
2762 .byte 0x0
2763 .uleb128 0x3
2764 .uleb128 0x8
2765 .uleb128 0xb
2766 .uleb128 0xb
2767 .uleb128 0x3e
2768 .uleb128 0xb
2769 .byte 0x0
2770 .byte 0x0

```

2771	.uleb128 0x5
2772	.uleb128 0xf
2773	.byte 0x0
2774	.uleb128 0xb
2775	.uleb128 0xb
2776	.uleb128 0x49
2777	.uleb128 0x13
2778	.byte 0x0
2779	.byte 0x0
2780	.uleb128 0x6
2781	.uleb128 0x17
2782	.byte 0x1
2783	.uleb128 0x1
2784	.uleb128 0x13
2785	.uleb128 0xb
2786	.uleb128 0xb
2787	.uleb128 0x3a
2788	.uleb128 0xb
2789	.uleb128 0x3b
2790	.uleb128 0xb
2791	.byte 0x0
2792	.byte 0x0
2793	.uleb128 0x7
2794	.uleb128 0xd
2795	.byte 0x0
2796	.uleb128 0x3
2797	.uleb128 0xe
2798	.uleb128 0x3a
2799	.uleb128 0xb
2800	.uleb128 0x3b
2801	.uleb128 0xb
2802	.uleb128 0x49
2803	.uleb128 0x13
2804	.byte 0x0
2805	.byte 0x0
2806	.uleb128 0x8
2807	.uleb128 0x1
2808	.byte 0x1
2809	.uleb128 0x1
2810	.uleb128 0x13
2811	.uleb128 0x49
2812	.uleb128 0x13
2813	.byte 0x0
2814	.byte 0x0
2815	.uleb128 0x9
2816	.uleb128 0x21
2817	.byte 0x0
2818	.uleb128 0x49
2819	.uleb128 0x13
2820	.uleb128 0x2f
2821	.uleb128 0xb
2822	.byte 0x0
2823	.byte 0x0
2824	.uleb128 0xa
2825	.uleb128 0x13
2826	.byte 0x1
2827	.uleb128 0x1
2828	.uleb128 0x13
2829	.uleb128 0x3
2830	.uleb128 0xe
2831	.uleb128 0xb
2832	.uleb128 0xb

2833	.uleb128 0x3a
2834	.uleb128 0xb
2835	.uleb128 0x3b
2836	.uleb128 0xb
2837	.byte 0x0
2838	.byte 0x0
2839	.uleb128 0xb
2840	.uleb128 0xd
2841	.byte 0x0
2842	.uleb128 0x3
2843	.uleb128 0xe
2844	.uleb128 0x3a
2845	.uleb128 0xb
2846	.uleb128 0x3b
2847	.uleb128 0xb
2848	.uleb128 0x49
2849	.uleb128 0x13
2850	.uleb128 0x38
2851	.uleb128 0xa
2852	.byte 0x0
2853	.byte 0x0
2854	.uleb128 0xc
2855	.uleb128 0xd
2856	.byte 0x0
2857	.uleb128 0x3
2858	.uleb128 0x8
2859	.uleb128 0x3a
2860	.uleb128 0xb
2861	.uleb128 0x3b
2862	.uleb128 0xb
2863	.uleb128 0x49
2864	.uleb128 0x13
2865	.uleb128 0x38
2866	.uleb128 0xa
2867	.byte 0x0
2868	.byte 0x0
2869	.uleb128 0xd
2870	.uleb128 0xf
2871	.byte 0x0
2872	.uleb128 0xb
2873	.uleb128 0xb
2874	.byte 0x0
2875	.byte 0x0
2876	.uleb128 0xe
2877	.uleb128 0x15
2878	.byte 0x1
2879	.uleb128 0x1
2880	.uleb128 0x13
2881	.uleb128 0x27
2882	.uleb128 0xc
2883	.uleb128 0x49
2884	.uleb128 0x13
2885	.byte 0x0
2886	.byte 0x0
2887	.uleb128 0xf
2888	.uleb128 0x5
2889	.byte 0x0
2890	.uleb128 0x49
2891	.uleb128 0x13
2892	.byte 0x0
2893	.byte 0x0
2894	.uleb128 0x10

2895	.uleb128 0x26
2896	.byte 0x0
2897	.uleb128 0x49
2898	.uleb128 0x13
2899	.byte 0x0
2900	.byte 0x0
2901	.uleb128 0x11
2902	.uleb128 0x16
2903	.byte 0x0
2904	.uleb128 0x3
2905	.uleb128 0xe
2906	.uleb128 0x3a
2907	.uleb128 0xb
2908	.uleb128 0x3b
2909	.uleb128 0x5
2910	.uleb128 0x49
2911	.uleb128 0x13
2912	.byte 0x0
2913	.byte 0x0
2914	.uleb128 0x12
2915	.uleb128 0x35
2916	.byte 0x0
2917	.uleb128 0x49
2918	.uleb128 0x13
2919	.byte 0x0
2920	.byte 0x0
2921	.uleb128 0x13
2922	.uleb128 0x13
2923	.byte 0x1
2924	.uleb128 0x1
2925	.uleb128 0x13
2926	.uleb128 0x3
2927	.uleb128 0xe
2928	.uleb128 0xb
2929	.uleb128 0xb
2930	.uleb128 0x3a
2931	.uleb128 0xb
2932	.uleb128 0x3b
2933	.uleb128 0x5
2934	.byte 0x0
2935	.byte 0x0
2936	.uleb128 0x14
2937	.uleb128 0xd
2938	.byte 0x0
2939	.uleb128 0x3
2940	.uleb128 0xe
2941	.uleb128 0x3a
2942	.uleb128 0xb
2943	.uleb128 0x3b
2944	.uleb128 0x5
2945	.uleb128 0x49
2946	.uleb128 0x13
2947	.uleb128 0x38
2948	.uleb128 0xa
2949	.byte 0x0
2950	.byte 0x0
2951	.uleb128 0x15
2952	.uleb128 0x13
2953	.byte 0x0
2954	.uleb128 0x3
2955	.uleb128 0xe
2956	.uleb128 0x3c

2957	.uleb128 0xc
2958	.byte 0x0
2959	.byte 0x0
2960	.uleb128 0x16
2961	.uleb128 0x13
2962	.byte 0x1
2963	.uleb128 0x1
2964	.uleb128 0x13
2965	.uleb128 0xb
2966	.uleb128 0xb
2967	.uleb128 0x3a
2968	.uleb128 0xb
2969	.uleb128 0x3b
2970	.uleb128 0xb
2971	.byte 0x0
2972	.byte 0x0
2973	.uleb128 0x17
2974	.uleb128 0x2e
2975	.byte 0x0
2976	.uleb128 0x3f
2977	.uleb128 0xc
2978	.uleb128 0x3
2979	.uleb128 0xe
2980	.uleb128 0x3a
2981	.uleb128 0xb
2982	.uleb128 0x3b
2983	.uleb128 0xb
2984	.uleb128 0x11
2985	.uleb128 0x1
2986	.uleb128 0x12
2987	.uleb128 0x1
2988	.uleb128 0x2001
2989	.uleb128 0x6
2990	.uleb128 0x40
2991	.uleb128 0xa
2992	.byte 0x0
2993	.byte 0x0
2994	.uleb128 0x18
2995	.uleb128 0x2e
2996	.byte 0x1
2997	.uleb128 0x1
2998	.uleb128 0x13
2999	.uleb128 0x3f
3000	.uleb128 0xc
3001	.uleb128 0x3
3002	.uleb128 0xe
3003	.uleb128 0x3a
3004	.uleb128 0xb
3005	.uleb128 0x3b
3006	.uleb128 0xb
3007	.uleb128 0x27
3008	.uleb128 0xc
3009	.uleb128 0x49
3010	.uleb128 0x13
3011	.uleb128 0x11
3012	.uleb128 0x1
3013	.uleb128 0x12
3014	.uleb128 0x1
3015	.uleb128 0x2001
3016	.uleb128 0x6
3017	.uleb128 0x40
3018	.uleb128 0xa

3019	.byte	0x0
3020	.byte	0x0
3021	.uleb128	0x19
3022	.uleb128	0x5
3023	.byte	0x0
3024	.uleb128	0x3
3025	.uleb128	0xe
3026	.uleb128	0x3a
3027	.uleb128	0xb
3028	.uleb128	0x3b
3029	.uleb128	0xb
3030	.uleb128	0x49
3031	.uleb128	0x13
3032	.uleb128	0x2
3033	.uleb128	0xa
3034	.byte	0x0
3035	.byte	0x0
3036	.uleb128	0x1a
3037	.uleb128	0x34
3038	.byte	0x0
3039	.uleb128	0x3
3040	.uleb128	0x8
3041	.uleb128	0x3a
3042	.uleb128	0xb
3043	.uleb128	0x3b
3044	.uleb128	0xb
3045	.uleb128	0x49
3046	.uleb128	0x13
3047	.uleb128	0x2
3048	.uleb128	0xa
3049	.byte	0x0
3050	.byte	0x0
3051	.uleb128	0x1b
3052	.uleb128	0x34
3053	.byte	0x0
3054	.uleb128	0x3
3055	.uleb128	0xe
3056	.uleb128	0x3a
3057	.uleb128	0xb
3058	.uleb128	0x3b
3059	.uleb128	0xb
3060	.uleb128	0x49
3061	.uleb128	0x13
3062	.uleb128	0x2
3063	.uleb128	0xa
3064	.byte	0x0
3065	.byte	0x0
3066	.uleb128	0x1c
3067	.uleb128	0x2e
3068	.byte	0x1
3069	.uleb128	0x1
3070	.uleb128	0x13
3071	.uleb128	0x3f
3072	.uleb128	0xc
3073	.uleb128	0x3
3074	.uleb128	0xe
3075	.uleb128	0x3a
3076	.uleb128	0xb
3077	.uleb128	0x3b
3078	.uleb128	0xb
3079	.uleb128	0x27
3080	.uleb128	0xc

```

3081 .uleb128 0x11
3082 .uleb128 0x1
3083 .uleb128 0x12
3084 .uleb128 0x1
3085 .uleb128 0x2001
3086 .uleb128 0x6
3087 .uleb128 0x40
3088 .uleb128 0xa
3089 .byte 0x0
3090 .byte 0x0
3091 .uleb128 0x1d
3092 .uleb128 0x5
3093 .byte 0x0
3094 .uleb128 0x3
3095 .uleb128 0x8
3096 .uleb128 0x3a
3097 .uleb128 0xb
3098 .uleb128 0x3b
3099 .uleb128 0xb
3100 .uleb128 0x49
3101 .uleb128 0x13
3102 .uleb128 0x2
3103 .uleb128 0xa
3104 .byte 0x0
3105 .byte 0x0
3106 .uleb128 0x1e
3107 .uleb128 0x1
3108 .byte 0x0
3109 .uleb128 0x3c
3110 .uleb128 0xb
3111 .uleb128 0x49
3112 .uleb128 0x13
3113 .byte 0x0
3114 .byte 0x0
3115 .uleb128 0x1f
3116 .uleb128 0x34
3117 .byte 0x0
3118 .uleb128 0x3
3119 .uleb128 0xe
3120 .uleb128 0x3a
3121 .uleb128 0xb
3122 .uleb128 0x3b
3123 .uleb128 0xb
3124 .uleb128 0x49
3125 .uleb128 0x13
3126 .uleb128 0x3f
3127 .uleb128 0xc
3128 .uleb128 0x3c
3129 .uleb128 0xc
3130 .byte 0x0
3131 .byte 0x0
3132 .byte 0x0
3133 .section .debug_pubnames,"",@progbits
3134 .4 byte 0x69
3135 .2 byte 0x2
3136 .4 byte $Ldebug_info0
3137 .4 byte 0xd3c
3138 .4 byte 0xb60
3139 .ascii "print_version\000"
3140 .4 byte 0xb76
3141 .ascii "print_usage\000"
3142 .4 byte 0xb8c

```

```

3143     .ascii    "is_pal\000"
3144     .4 byte   0xbe0
3145     .ascii    "chomp\000"
3146     .4 byte   0xc08
3147     .ascii    "handle_line\000"
3148     .4 byte   0xc54
3149     .ascii    "handle\000"
3150     .4 byte   0xcc8
3151     .ascii    "main\000"
3152     .4 byte   0x0
3153     .section   .debug_aranges,"",@progbits
3154     .4 byte   0x1c
3155     .2 byte   0x2
3156     .4 byte   $Ldebug_info0
3157     .byte     0x4
3158     .byte     0x0
3159     .2 byte   0x0
3160     .2 byte   0x0
3161     .4 byte   $Ltext0
3162     .4 byte   $Ltext0-$Ltext0
3163     .4 byte   0x0
3164     .4 byte   0x0
3165     .section   .debug_str,"MS",@progbits,1
3166 $LC74:
3167     .ascii    "_ext\000"
3168 $LC63:
3169     .ascii    "_size\000"
3170 $LC145:
3171     .ascii    "timer_t\000"
3172 $LC161:
3173     .ascii    "pthread_mutex_t\000"
3174 $LC203:
3175     .ascii    "__pthread_rwlockattr_t\000"
3176 $LC157:
3177     .ascii    "__pthread_attr_t\000"
3178 $LC97:
3179     .ascii    "uint32_t\000"
3180 $LC165:
3181     .ascii    "ptm_interlock\000"
3182 $LC164:
3183     .ascii    "ptm_lock\000"
3184 $LC34:
3185     .ascii    "short unsigned int\000"
3186 $LC68:
3187     .ascii    "_lbfsize\000"
3188 $LC152:
3189     .ascii    "pthread_queue_t\000"
3190 $LC30:
3191     .ascii    "unsigned char\000"
3192 $LC185:
3193     .ascii    "__pthread_once_t\000"
3194 $LC132:
3195     .ascii    "mode_t\000"
3196 $LC147:
3197     .ascii    "useconds_t\000"
3198 $LC210:
3199     .ascii    "ptb_waiters\000"
3200 $LC205:
3201     .ascii    "ptr-private\000"
3202 $LC172:
3203     .ascii    "ptma-private\000"
3204 $LC80:

```



```

3205     .ascii  "off_t\000"
3206 $LC199:
3207     .ascii  "ptr_nreaders\000"
3208 $LC201:
3209     .ascii  "ptr_private\000"
3210 $LC64:
3211     .ascii  "__sbuf\000"
3212 $LC72:
3213     .ascii  "_seek\000"
3214 $LC48:
3215     .ascii  "__in_port_t\000"
3216 $LC60:
3217     .ascii  "size_t\000"
3218 $LC118:
3219     .ascii  "longlong_t\000"
3220 $LC129:
3221     .ascii  "id_t\000"
3222 $LC71:
3223     .ascii  "_read\000"
3224 $LC119:
3225     .ascii  "u_longlong_t\000"
3226 $LC50:
3227     .ascii  "__off_t\000"
3228 $LC195:
3229     .ascii  "ptr_magic\000"
3230 $LC232:
3231     .ascii  "line\000"
3232 $LC204:
3233     .ascii  "ptr_magic\000"
3234 $LC233:
3235     .ascii  "out_f\000"
3236 $LC187:
3237     .ascii  "pto_done\000"
3238 $LC85:
3239     .ascii  "mips_fpreg_t\000"
3240 $LC56:
3241     .ascii  "__fsfilcnt_t\000"
3242 $LC146:
3243     .ascii  "suseconds_t\000"
3244 $LC123:
3245     .ascii  "fsfilcnt_t\000"
3246 $LC114:
3247     .ascii  "cpuid_t\000"
3248 $LC65:
3249     .ascii  "__sFILE\000"
3250 $LC57:
3251     .ascii  "__mbstate8\000"
3252 $LC83:
3253     .ascii  "mips_ureg_t\000"
3254 $LC153:
3255     .ascii  "ptqh_first\000"
3256 $LC108:
3257     .ascii  "u_int\000"
3258 $LC92:
3259     .ascii  "int8_t\000"
3260 $LC148:
3261     .ascii  "__fd_mask\000"
3262 $LC58:
3263     .ascii  "__mbstateL\000"
3264 $LC191:
3265     .ascii  "pts_spin\000"
3266 $LC110:

```

```

3267     .ascii    "uchar\000"
3268 $LC121:
3269     .ascii    "blksize_t\000"
3270 $LC173:
3271     .ascii    "pthread_cond_t\000"
3272 $LC168:
3273     .ascii    "ptm_private\000"
3274 $LC171:
3275     .ascii    "ptma_magic\000"
3276 $LC222:
3277     .ascii    "div_t\000"
3278 $LC225:
3279     .ascii    "qdiv_t\000"
3280 $LC87:
3281     .ascii    "psize_t\000"
3282 $LC209:
3283     .ascii    "ptb_lock\000"
3284 $LC70:
3285     .ascii    "_close\000"
3286 $LC198:
3287     .ascii    "ptr_wblocked\000"
3288 $LC130:
3289     .ascii    "ino_t\000"
3290 $LC61:
3291     .ascii    "fpos_t\000"
3292 $LC40:
3293     .ascii    "__uint64_t\000"
3294 $LC239:
3295     .ascii    "argc\000"
3296 $LC134:
3297     .ascii    "pid_t\000"
3298 $LC89:
3299     .ascii    "vsize_t\000"
3300 $LC99:
3301     .ascii    "uint64_t\000"
3302 $LC149:
3303     .ascii    "fd_set\000"
3304 $LC240:
3305     .ascii    "argv\000"
3306 $LC69:
3307     .ascii    "_cookie\000"
3308 $LC234:
3309     .ascii    "handle\000"
3310 $LC224:
3311     .ascii    "lldiv_t\000"
3312 $LC84:
3313     .ascii    "long unsigned int\000"
3314 $LC183:
3315     .ascii    "ptca_private\000"
3316 $LC241:
3317     .ascii    "__sF\000"
3318 $LC51:
3319     .ascii    "__pid_t\000"
3320 $LC77:
3321     .ascii    "_blksize\000"
3322 $LC62:
3323     .ascii    "_base\000"
3324 $LC131:
3325     .ascii    "key_t\000"
3326 $LC117:
3327     .ascii    "qaddr_t\000"
3328 $LC231:

```

```

3329     .ascii    "handle_line\000"
3330 $LC236:
3331     .ascii    "output_file_name\000"
3332 $LC36:
3333     .ascii    "__uint32_t\000"
3334 $LC206:
3335     .ascii    "pthread_barrier_t\000"
3336 $LC179:
3337     .ascii    "ptc_private\000"
3338 $LC221:
3339     .ascii    "quot\000"
3340 $LC126:
3341     .ascii    "dev_t\000"
3342 $LC141:
3343     .ascii    "clock_t\000"
3344 $LC135:
3345     .ascii    "lwpid_t\000"
3346 $LC75:
3347     .ascii    "_ubuf\000"
3348 $LC76:
3349     .ascii    "_nbuf\000"
3350 $LC245:
3351     .ascii    "tp0.c\000"
3352 $LC32:
3353     .ascii    "short int\000"
3354 $LC100:
3355     .ascii    "u_int8_t\000"
3356 $LC180:
3357     .ascii    "pthread_condattr_t\000"
3358 $LC98:
3359     .ascii    "int64_t\000"
3360 $LC139:
3361     .ascii    "uid_t\000"
3362 $LC217:
3363     .ascii    "ptba_magic\000"
3364 $LC39:
3365     .ascii    "long long int\000"
3366 $LC216:
3367     .ascii    "__pthread_barrierattr_st\000"
3368 $LC202:
3369     .ascii    "pthread_rwlockattr_t\000"
3370 $LC127:
3371     .ascii    "fixpt_t\000"
3372 $LC125:
3373     .ascii    "daddr_t\000"
3374 $LC238:
3375     .ascii    "main\000"
3376 $LC41:
3377     .ascii    "long long unsigned int\000"
3378 $LC247:
3379     .ascii    "GNU C 3.3.3 (NetBSD nb3 20040520) -g\000"
3380 $LC54:
3381     .ascii    "__uid_t\000"
3382 $LC79:
3383     .ascii    "FILE\000"
3384 $LC33:
3385     .ascii    "__uint16_t\000"
3386 $LC133:
3387     .ascii    "nlink_t\000"
3388 $LC138:
3389     .ascii    "swblk_t\000"
3390 $LC151:

```

```

3391      .ascii  "pthread_spin_t\000"
3392 $LC137:
3393      .ascii  "segsz_t\000"
3394 $LC215:
3395      .ascii  "pthread_barrierattr_t\000"
3396 $LC140:
3397      .ascii  "dtime_t\000"
3398 $LC45:
3399      .ascii  "char\000"
3400 $LC223:
3401      .ascii  "ldiv_t\000"
3402 $LC182:
3403      .ascii  "ptca_magic\000"
3404 $LC86:
3405      .ascii  "paddr_t\000"
3406 $LC244:
3407      .ascii  "_tolower_tab_\000"
3408 $LC212:
3409      .ascii  "ptb_curcount\000"
3410 $LC96:
3411      .ascii  "int32_t\000"
3412 $LC184:
3413      .ascii  "pthread_once_t\000"
3414 $LC88:
3415      .ascii  "vaddr_t\000"
3416 $LC218:
3417      .ascii  "ptba_private\000"
3418 $LC159:
3419      .ascii  "pta_flags\000"
3420 $LC226:
3421      .ascii  "print_version\000"
3422 $LC93:
3423      .ascii  "uint8_t\000"
3424 $LC176:
3425      .ascii  "ptc_lock\000"
3426 $LC81:
3427      .ascii  "mips_reg_t\000"
3428 $LC230:
3429      .ascii  "word\000"
3430 $LC197:
3431      .ascii  "ptr_rblocked\000"
3432 $LC124:
3433      .ascii  "caddr_t\000"
3434 $LC189:
3435      .ascii  "__pthread_spinlock_st\000"
3436 $LC214:
3437      .ascii  "ptb_private\000"
3438 $LC194:
3439      .ascii  "__pthread_rwlock_st\000"
3440 $LC175:
3441      .ascii  "ptc_magic\000"
3442 $LC112:
3443      .ascii  "uint\000"
3444 $LC181:
3445      .ascii  "__pthread_condattr_st\000"
3446 $LC192:
3447      .ascii  "pts_flags\000"
3448 $LC178:
3449      .ascii  "ptc_mutex\000"
3450 $LC242:
3451      .ascii  "__pthread_st\000"
3452 $LC66:

```

```

3453     .ascii    "_flags\000"
3454 $LC193:
3455     .ascii    "pthread_rwlock_t\000"
3456 $LC42:
3457     .ascii    "__intptr_t\000"
3458 $LC208:
3459     .ascii    "ptb-magic\000"
3460 $LC94:
3461     .ascii    "int16_t\000"
3462 $LC120:
3463     .ascii    "blkcnt_t\000"
3464 $LC220:
3465     .ascii    "wchar_t\000"
3466 $LC186:
3467     .ascii    "pto-mutex\000"
3468 $LC188:
3469     .ascii    "pthread_spinlock_t\000"
3470 $LC160:
3471     .ascii    "pta-private\000"
3472 $LC29:
3473     .ascii    "signed char\000"
3474 $LC162:
3475     .ascii    "__pthread_mutex_st\000"
3476 $LC49:
3477     .ascii    "__mode_t\000"
3478 $LC90:
3479     .ascii    "register_t\000"
3480 $LC167:
3481     .ascii    "ptm-blocked\000"
3482 $LC144:
3483     .ascii    "clockid_t\000"
3484 $LC213:
3485     .ascii    "ptb-generation\000"
3486 $LC37:
3487     .ascii    "unsigned int\000"
3488 $LC116:
3489     .ascii    "quad_t\000"
3490 $LC109:
3491     .ascii    "u_long\000"
3492 $LC200:
3493     .ascii    "ptr-writer\000"
3494 $LC196:
3495     .ascii    "ptr-interlock\000"
3496 $LC103:
3497     .ascii    "u_int64_t\000"
3498 $LC169:
3499     .ascii    "pthread_mutexattr_t\000"
3500 $LC166:
3501     .ascii    "ptm-owner\000"
3502 $LC219:
3503     .ascii    "pthread_key_t\000"
3504 $LC246:
3505     .ascii    "/tmp\000"
3506 $LC113:
3507     .ascii    "ulong\000"
3508 $LC106:
3509     .ascii    "u_char\000"
3510 $LC53:
3511     .ascii    "__socklen_t\000"
3512 $LC142:
3513     .ascii    "ssize_t\000"
3514 $LC174:

```

```

3515         .ascii    "--pthread_cond_st\000"
3516 $LC38:
3517         .ascii    "--int64_t\000"
3518 $LC155:
3519         .ascii    "pthread_t\000"
3520 $LC67:
3521         .ascii    "_file\000"
3522 $LC104:
3523         .ascii    "in_addr_t\000"
3524 $LC27:
3525         .ascii    "--int8_t\000"
3526 $LC55:
3527         .ascii    "--fsblkcnt_t\000"
3528 $LC128:
3529         .ascii    "gid_t\000"
3530 $LC122:
3531         .ascii    "fsblkcnt_t\000"
3532 $LC91:
3533         .ascii    "--cpu_simple_lock_t\000"
3534 $LC102:
3535         .ascii    "u_int32_t\000"
3536 $LC78:
3537         .ascii    "_offset\000"
3538 $LC59:
3539         .ascii    "--mbstate_t\000"
3540 $LC154:
3541         .ascii    "ptqh_last\000"
3542 $LC111:
3543         .ascii    "ushort\000"
3544 $LC43:
3545         .ascii    "--uintptr_t\000"
3546 $LC46:
3547         .ascii    "--gid_t\000"
3548 $LC35:
3549         .ascii    "--int32_t\000"
3550 $LC211:
3551         .ascii    "ptb_initcount\000"
3552 $LC170:
3553         .ascii    "--pthread_mutexattr_st\000"
3554 $LC73:
3555         .ascii    "_write\000"
3556 $LC28:
3557         .ascii    "--uint8_t\000"
3558 $LC150:
3559         .ascii    "fds_bits\000"
3560 $LC158:
3561         .ascii    "pta_magic\000"
3562 $LC227:
3563         .ascii    "print_usage\000"
3564 $LC243:
3565         .ascii    "--pthread_spin_st\000"
3566 $LC235:
3567         .ascii    "input_file_name\000"
3568 $LC95:
3569         .ascii    "uint16_t\000"
3570 $LC82:
3571         .ascii    "long_int\000"
3572 $LC156:
3573         .ascii    "pthread_attr_t\000"
3574 $LC44:
3575         .ascii    "--caddr_t\000"
3576 $LC105:

```

```

3577      .ascii  "in_port_t\000"
3578 $LC237:
3579      .ascii  "in_f\000"
3580 $LC207:
3581      .ascii  "__pthread_barrier_st\000"
3582 $LC101:
3583      .ascii  "u_int16_t\000"
3584 $LC143:
3585      .ascii  "time_t\000"
3586 $LC229:
3587      .ascii  "chomp\000"
3588 $LC163:
3589      .ascii  "ptm_magic\000"
3590 $LC115:
3591      .ascii  "u_quad_t\000"
3592 $LC228:
3593      .ascii  "is_pal\000"
3594 $LC190:
3595      .ascii  "pts_magic\000"
3596 $LC47:
3597      .ascii  "__in_addr_t\000"
3598 $LC136:
3599      .ascii  "rlim_t\000"
3600 $LC52:
3601      .ascii  "__sa_family_t\000"
3602 $LC107:
3603      .ascii  "u_short\000"
3604 $LC31:
3605      .ascii  "__int16_t\000"
3606 $LC177:
3607      .ascii  "ptc_waiters\000"
3608      .ident  "GCC: (GNU) 3.3.3 (NetBSD nb3 20040520)"

```

tp0.s

## 4. Conclusiones

Al final de este proyecto más allá de la simpleza del mismo, pudimos observar los usos de cada distinta herramienta. Observamos como LaTeX sirve para escribir documentos que particularmente incluyan código de programas, puesto que su estructura y posibilidad de personalizar el formato hace más facil su lectura. Vimos como gxemul nos posibilita levantar una imagen virtual de otro sistema operativo dentro de nuestra propia máquina. Esta imagen sea NetBSD que utilizamos para poder trabajar con otras arquitecturas que nuestro sistema operativo no permitía, como en particular assembly y la obtención del código en un '.s'. Cabe resaltar esta particularidad de NetBSD ya que es la base fundamental para la realización del siguiente trabajo práctico, que constará en trabajar con el código asm de este trabajo, y solo podemos hacerlo mediante la máquina virtual