
ORGANIZACIÓN DE COMPUTADORAS

INFORME TRABAJO PRÁCTICO 0

Alumnos

93198 - Peña, Maximiliano
maxipenia@gmail.com

93665 - Poggio, Demian
demianpoggio@gmail.com

95505 - Iogha, Octavio
octaviomdq93@gmail.com

Fecha de Vencimiento

Martes 27 de Septiembre

1. Objetivos

El objetivo del trabajo práctico es familiarizarnos con las herramientas que necesitaremos para trabajar a lo largo de la cursada. Estas herramientas son:

- Compilador GCC
- Shell de Unix
- Emulador GXEmul

2. Introducción Teórica

Para poner en práctica estas herramientas, se debe implementar un programa en el lenguaje de programación C que permita, ingresando diversos parametros, generar imágenes en formato pgm correspondientes al conjunto de Julia y sus vecindades. Los parámetros para la generación de la imagen serán ingresados al llamar al programa. A continuación se explican brevemente los conceptos necesarios para llevar a cabo esta tarea:

2.1. Shell de Unix

Un shell es un programa intérprete de comandos. Los comandos utilizados en el presente trabajo son los siguientes:

2.2. Compilador GCC

GCC es un compilador gratuito, Open Source y multiplataforma que permite compilar código C. El mismo es un compilador estándar, es decir, que respeta la norma ISO C99 o ANSI dependiendo de su versión, por lo cual se utilizan las bibliotecas estándar para trabajar con el mismo. En esta asignatura el mismo es de gran utilidad dado que permite tener acceso al código *assembly* equivalente a nuestro programa en C. A continuación se detallan brevemente algunos parámetros que podemos darle al GCC:

- **-Wall:** Activa todos los warnings.
- **-o file** Permite cambiar el nombre del archivo compilado generado.
- **-O0:** Desactiva las optimizaciones, lo cual resulta especialmente útil, porque genera código fácilmente entendible en assembly en comparación al código C.
- **-O3:** Activa todas las optimizaciones, lo cual deseamos evitar, porque el código generado en assembly no se traduce de forma simple al código C correspondiente.
- **-g:** Genera código agregando información de debugging.
- **-S:** Detiene el compilador luego de generar el código assembly.

- **-mrnames(solo para MIPS):** Indica al compilador que genera la salida utilizando nombre de registro en lugar de número de registro.

Para generar el archivo “main.s” con el código assembly se debe ejecutar el siguiente comando:

```
gcc -std=c99 -S -mrnames -O0 -g tp0.c
```

Para compilar el código se debe ejecutar el siguiente comando:

```
gcc -std=c99 -O0 -g tp0.c -o tp0 -lm
```

2.3. Emulador GXEmul

Es un emulador gratuito y Open Source de la arquitectura de computadores MIPS. Tiene como ventajas que puede correr algunos sistemas operativos sin modificaciones como netBSD, su portabilidad y velocidad de emulación dado que realiza la traducción binaria en tiempo real.

3. Diseño e implementación

Para la implementación del programa se creó el archivo:

- main.c: Contiene la implementación del software.

El programa utiliza los parámetros recibidos y comienza a realizar los cálculos necesarios para generar la imagen del conjunto de Julia. Una vez que realiza los cálculos necesarios genera el documento pgm en donde guarda la imagen.

3.1. Compilación

Para compilar el programa basta con ejecutar

```
gcc -std=c99 -O0 -g tp0.c -o <nombre salida> -lm
```

3.2. Modo de Operación

La aplicación desarrollada recibe los parámetros deseados y los utiliza para la generación de la imagen. Los mismos se describen a continuación:

- `tp0 -h -V -c <a+bi> -H <float> -w <float> -o <out file>`
- **-V** Imprime la versión y finaliza.
- **-h** Imprime información y finaliza.
- **-c** Setea el centro de la imagen.

- **-H** Setea el alto del rectangulo. Valor por defecto=4.
- **-w** Setea el ancho del rectangulo. Valor por defecto=4.
- **-o** Setea el archivo de salida.

Para ejecutarlo se debe hacer:

```
$ ./tp0 -o uno.pgm
```

o

```
./tp0 -C -1.125-0.21650635094611i -o dos.pgm
```

3.3. Ejecución de prueba

Primero, usamos la opcion -h para ver el mensaje de ayuda:

Usage:

```
tp0 -h -V -c <a+bi> -C <a+bi> -H <float> -w <float> -o <out_file>
```

Options:

```
-V      Imprime la version y finaliza.
-h      Imprime esta informacion y finaliza.
-c      Setea el centro de la imagen.
-H      Setea el alto del rectangulo. Valor por defecto=4
-w      Setea el ancho del rectangulo. Valor por defecto=4
-o      Setea el archivo de salida
-C      Setea la constante del algoritmo. Valor por defecto= 0.285+0.01i
```

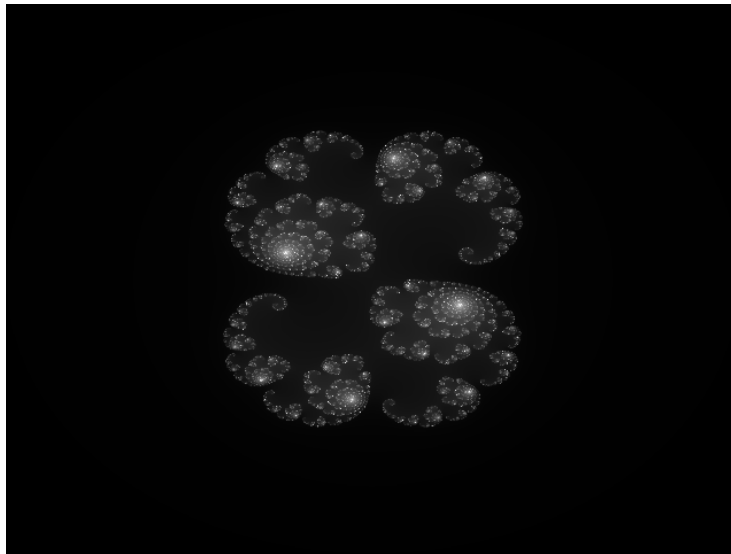
Examples:

```
tp0 -c +0.282-0.01i -w 0.005 -H 0.005 -o dos.pgm
```

Luego, hacemos una corrida de prueba con la linea:

```
$ ./tp0 -o uno.pgm
```

Y el resultado de la ejecución fue:

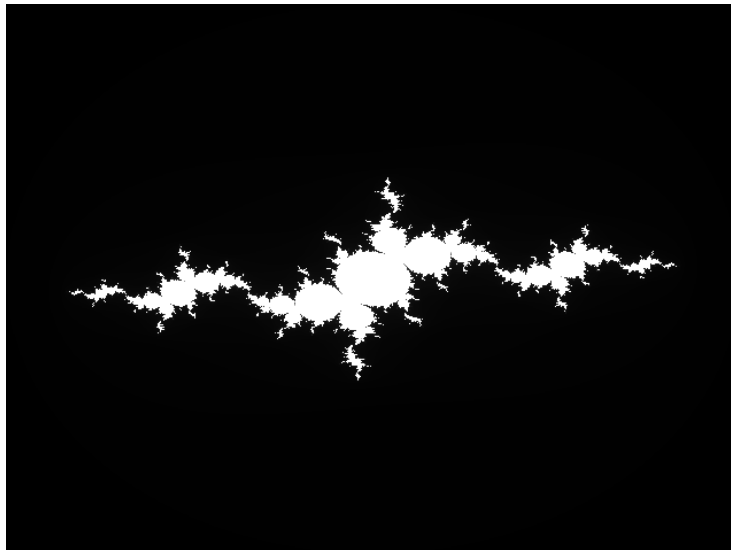


Los resultados coinciden con los esperados, igual al ejemplo del enunciado.

Probamos con otro caso:

```
$ ./tp0 -C -1.125-0.21650635094611i -o dos.pgm
```

Y el resultado de la ejecución fue:



Los resultados coinciden con los esperados, aunque invertidos respecto del ejemplo del enunciado.

4. Corridas de prueba

4.1. Pruebas

4.1.1. Imagen 1x1

```
./tp0 -c 0.01+0i -r 1x1 -o -  
P2  
1 1  
255  
19
```

La salida es la esperada.

4.1.2. Imagen 1x1 con punto que no pertenece al conjunto

```
./tp0 -c 10+0i -r 1x1 -o -  
P2  
1 1  
255  
0
```

La salida es la esperada.

5. Código fuente

```
1 #include <stdio.h>  
2 #include <stdlib.h>  
3 #include <string.h>  
4 #include <math.h>  
5 #define ARG_ERR -1  
6 #define FILE_ERROR -2  
7 #define ERR_GRAL 1  
8 #define ERR_MEM -3  
9 typedef struct{  
10     double real;  
11     float imag;  
12 }numcomplex;  
13  
14 void armar_headerPGM(FILE* salida, int alto, int ancho){  
15     fputs("P2 \n", salida);  
16     fputs("#TP0 Vecindades de Julia \n", salida);  
17     char alto_str[10];  
18     char ancho_str[10];  
19     char concat[20];  
20     sprintf(alto_str, "%d \n", alto);  
21     sprintf(ancho_str, "%d", ancho);  
22     strcpy(concat, ancho_str);  
23     strcat(concat, " ");  
24     strcat(concat, alto_str);
```

```

25  fputs(concat, salida);
26  fputs("255 \n", salida);
27 }
28
29 void armar_imagenPGM(FILE* salida, int** matrix_PGM, int
    alto, int ancho){
30  for(int im = 0; im<alto; im++){
31      for(int re=0; re<ancho; re++){
32          fprintf(salida, "%d ", matrix_PGM[im][re]);
33      }
34  }
35  fprintf(salida, "\n");
36 }
37 void imprimir_complejo(numcomplex c){
38  printf(" %f %tfi \n", c.real, c.imag);
39 }
40
41 double abs_cplx(numcomplex a){
42  return sqrt(pow(a.real, 2) + pow(a.imag, 2));
43 }
44
45 void sqr_cplx(numcomplex* a){
46  double aux = a->imag;
47  a->imag = 2 * a->real * a->imag;
48  a->real=pow(a->real, 2) - pow(aux, 2);
49 }
50
51 void imprimir_error(int status){
52  switch(status){
53      case ARGERR:
54          printf("Debe ingresar correctamente los
              argumentos. Abortando ejecucion\n");
55          exit(ARGERR);
56          break;
57      case FILEERROR:
58          printf("La ruta de archivo ingresada no es valida.
              Abortando ejecucion\n");
59          exit(FILEERROR);
60      case ERRMEM:
61          printf("No se ha podido reservar la memoria
              necesaria. Abortando ejecucion\n");
62          exit(ERRMEM);
63      default:
64          printf("Error no contemplado. Abortando Ejecucion
              \n");
65  }  exit(status);
66 }
67
68 int** create_matrix(int alto, int ancho, int status){
69  int** matrix_PGM= malloc(alto * sizeof(*matrix_PGM));

```

```

70
71     if (matrix_PGM == NULL){
72         status = ERRMEM;
73         imprimir_error(status);
74     }
75     for(int i = 0; i < alto; i++){
76         matrix_PGM[i] = malloc(ancho*
77             sizeof(*matrix_PGM[i]));
78         if (matrix_PGM[i] == NULL){
79             status = ERRMEM;
80             imprimir_error(status);
81         }
82     }
83     return matrix_PGM;
84 }
85 void generate_julia(int** matrix_PGM,int ancho,int
86     alto,double w, double H,numcomplex constant,
87     numcomplex center){
88     double aux_im;
89     int n = 0;
90     double xmin=-w/2;
91     double xmax=w/2;
92     double ymin=-H/2;
93     double ymax=H/2;
94     double deltaX,deltaY;
95     if(ancho!=1){
96         deltaX=(xmax-xmin)/(ancho-1);
97     }
98     else{
99         deltaX=(xmax-xmin);
100     }
101     if(alto!=1){
102         deltaY=(ymax-ymin)/(alto-1);
103     }
104     else{
105         deltaY=(ymax-ymin);
106     }
107     //if (ancho == 1) deltaX = 1;
108     //if (alto == 1) deltaY = 0;
109     for (int im=0; im<alto;im++){
110         aux_im=((ymax - deltaY/2 - (im)*deltaY)) +
111             center.imag;
112         numcomplex zeta;
113         for(int re=0;re<ancho;re++){
114             zeta.real=(xmin + deltaX/2 + (re)*deltaX) +
115                 center.real;
116             zeta.imag=aux_im;
117         }
118     }

```



```

115     while( abs_cplx(zeta) < 2 && n<255){
116
117         sqr_cplx(&zeta);
118         zeta.real+=constant.real;
119         zeta.imag+=constant.imag;
120         n++;
121     }
122     matrix_PGM[im][re]= n;
123     n=0;
124 }
125 }
126 }
127
128 void free_matrix(int** matrix_PGM, int alto, int ancho){
129     for (int i = 0; i < alto; i++){
130         free(matrix_PGM[i]);
131     }
132     free(matrix_PGM);
133 }
134
135 int main(int argc, char *argv[])
136 {
137     int status = 0;
138     int alto = 480;
139     int ancho = 640;
140     numcomplex constant;
141     numcomplex center;
142     center.real=0;
143     center.imag=0;
144     constant.real=0.285;
145     constant.imag=-0.01;
146     char *auxc;
147     char *auxc2;
148     double H=4,w=4;
149     FILE *salida = stdout;
150
151     if (argc > 1 ) {
152         for(int i=1 ; i<argc ; i+=2){
153             if(argv[i][0] == '-') {
154
155                 switch (argv[i][1]) {
156                     case 'h': printf( "Usage:\n  tp0 -h -V -c
157                               <a+bi> -C <a+bi> -H <float> -w <float> -o
158                               <out_file> -\n"
159                               " Options:\n"
160                               "   -V\t Imprime la version y
161                               finaliza.\n"
162                               "   -h\t Imprime esta informacion y
163                               finaliza.\n"
164                               "   -c\t Setea el centro de la imagen.\n"

```

```

161      -H\t      Setea el alto del rectangulo.
          Valor por defecto=4\n"
162      -w\t      Setea el ancho del rectangulo.
          Valor por defecto=4\n"
163      -o\t      Setea el archivo de salida"
164      -C\t      Setea la constante del
          algoritmo. Valor por defecto=
          0.285+0.01i\n"
165      "Examples:\n  tp0 -c +0.282-0.01i -w 0.005
          -H 0.005 -o dos.pgm\n");
166      //texto con ayuda a completar//
167      return 0;
168      break;
169  case 'V': printf("Conjunto de Julia\nv1.0\n");
170      return 0;
171      break;
172  case 'r':
173      auxc=strtok(argv[i+1],"xX");
174      if (auxc == NULL){
175          status = ARG_ERR;
176          break;
177      }
178      ancho=atoi(auxc);
179
180      auxc=strtok(NULL, " ");
181      if (auxc == NULL) {
182          status = ARG_ERR;
183          break;
184      }
185      alto=atoi(auxc);
186      if(ancho==0 || alto==0){
187          status = ARG_ERR;
188          break;
189      }
190      break;
191  case 'C':
192      auxc=argv[i+1];
193      if(auxc == NULL){
194          status=ARG_ERR;
195          break;
196      }
197      constant.real=atof(auxc);
198      if(argv[i+1][0]=='-' || argv[i+1][0]=='+'){
          //si el primer numero tiene signo,
          lo saltea
199      auxc=strpbrk(auxc+1,"-+");
200      }
201      else auxc=strpbrk(auxc,"-+");
202      auxc2 = strpbrk(auxc, "i");
203      if (!auxc2){

```

```

204         status = ARGERR;
205         break;
206     }
207     constant.imag=atof(auxc);
208     break;
209 case 'c':
210
211     auxc=argv[i+1];
212     if (auxc == NULL){
213         status=ARGERR;
214         break;
215     }
216     center.real=atof(auxc);
217     //toma la parte real
218     if (argv[i+1][0]=='-' || argv[i+1][0]=='+') {
219         //si el primer numero tiene signo,
220         lo saltea
221         auxc=strpbrk(auxc+1,"-+");
222     }
223     else{
224         auxc=strpbrk(auxc,"-+");
225     }
226     auxc2 = strpbrk(auxc, "i");
227     if (!auxc2){
228         status = ARGERR;
229         break;
230     }
231     center.imag=atof(auxc);
232     //toma la parte imaginaria
233     break;
234 case 'H':
235     if (argv[i+1] == NULL){
236         status=ARGERR;
237         break;
238     }
239     H=atof(argv[i+1]);
240     break;
241 case 'o':
242     if (argv[i+1] == NULL){
243         status=ARGERR;
244         break;
245     }
246     if (strcmp(argv[i+1],"-") == 0){
247         break;
248     };
249
250     salida = fopen(argv[i+1], "w");
251     if (salida == NULL){
252         status=FILE.ERROR;
253         break;

```

```

250     }
251     break;
252     case 'w':
253         if (argv[i+1] == NULL){
254             status=ARG_ERR;
255             break;
256         }
257         w=atof(argv[i+1]);
258         break;
259     default: printf("Argumento desconocido: prueba
                con -h para ver la ayuda.\n");
260 }
261 }
262 else{
263     printf("Error! Formato desconocido. Prueba con
                -h para ver la ayuda. \n");
264     return ERR_GRAL;
265     //return error de argumento
266 }
267 }
268 }
269 else{
270     printf("Se corra el programa con los valores por
                DEFAULT. \n");
271 }
272 if (status != 0) imprimir_error(status);
273 armar_headerPGM(salida, alto, ancho);
274
275 int** matrix_PGM;
276 matrix_PGM = create_matrix(alto, ancho, status);
277
278 generate_julia(matrix_PGM, ancho, alto, w, H, constant, center);
279 armar_imagenPGM(salida, matrix_PGM, alto, ancho);
280
281 free_matrix(matrix_PGM, alto, ancho);
282 fclose(salida);
283 return 0;
284 }

```

tp0.c

6. Código MIPS

```

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 .file 11 "/usr/include/math.h"
23 .rdata
24 .align 2
25 $LC0:
26 .ascii "P2 \n\000"
27 .align 2
28 $LC1:
29 .ascii "#TP0 Vecindades de Julia \n\000"
30 .align 2
31 $LC2:
32 .ascii "%d \n\000"
33 .align 2
34 $LC3:
35 .ascii "%d\000"
36 .align 2
37 $LC4:
38 .ascii " \000"
39 .align 2
40 $LC5:
41 .ascii "255 \n\000"
42 .text
43 .align 2
44 .globl armar_headerPGM
45 $LFB29:
46 .loc 1 14 0
47 .ent armar_headerPGM
48 armar_headerPGM:
49 .frame $fp,96,$ra # vars= 56, regs= 3/0, args= 16,
    extra= 8
50 .mask 0xd0000000,-8
51 .fmask 0x00000000,0
52 .set noreorder
53 .cpload $t9
54 .set reorder
55 subu $sp,$sp,96

```

```

56     .cprestore 16
57 $LCFI0:
58     sw     $ra,88($sp)
59 $LCFI1:
60     sw     $fp,84($sp)
61 $LCFI2:
62     sw     $gp,80($sp)
63 $LCFI3:
64     move   $fp,$sp
65 $LCFI4:
66     sw     $a0,96($fp)
67     sw     $a1,100($fp)
68     sw     $a2,104($fp)
69     .loc   1 15 0
70 $LBB2:
71     la     $a0,$LC0
72     lw     $a1,96($fp)
73     la     $t9,fputs
74     jal    $ra,$t9
75     .loc   1 16 0
76     la     $a0,$LC1
77     lw     $a1,96($fp)
78     la     $t9,fputs
79     jal    $ra,$t9
80     .loc   1 20 0
81     addu   $a0,$fp,24
82     la     $a1,$LC2
83     lw     $a2,100($fp)
84     la     $t9,sprintf
85     jal    $ra,$t9
86     .loc   1 21 0
87     addu   $v0,$fp,40
88     move   $a0,$v0
89     la     $a1,$LC3
90     lw     $a2,104($fp)
91     la     $t9,sprintf
92     jal    $ra,$t9
93     .loc   1 22 0
94     addu   $v0,$fp,56
95     addu   $v1,$fp,40
96     move   $a0,$v0
97     move   $a1,$v1
98     la     $t9,strcpy
99     jal    $ra,$t9
100    .loc   1 23 0
101    addu   $v0,$fp,56
102    move   $a0,$v0
103    la     $a1,$LC4
104    la     $t9, strcat
105    jal    $ra,$t9

```

```

106  .loc 1 24 0
107  addu $v0,$fp,56
108  move $a0,$v0
109  addu $a1,$fp,24
110  la $t9, strcat
111  jal $ra,$t9
112  .loc 1 25 0
113  addu $v0,$fp,56
114  move $a0,$v0
115  lw $a1,96($fp)
116  la $t9, fputs
117  jal $ra,$t9
118  .loc 1 26 0
119  la $a0,$LC5
120  lw $a1,96($fp)
121  la $t9, fputs
122  jal $ra,$t9
123  .loc 1 27 0
124  move $sp,$fp
125  lw $ra,88($sp)
126  lw $fp,84($sp)
127  addu $sp,$sp,96
128  j $ra
129 $LBE2:
130 .end armar_headerPGM
131 $LFE29:
132 .size armar_headerPGM, .-armar_headerPGM
133 .rdata
134 .align 2
135 $LC6:
136 .ascii "%d \000"
137 .align 2
138 $LC7:
139 .ascii "\n\000"
140 .text
141 .align 2
142 .globl armar_imagenPGM
143 $LFB31:
144 .loc 1 29 0
145 .ent armar_imagenPGM
146 armar_imagenPGM:
147 .frame $fp,48,$ra # vars= 8, regs= 3/0, args= 16,
    extra= 8
148 .mask 0xd0000000,-8
149 .fmask 0x00000000,0
150 .set noreorder
151 .cpload $t9
152 .set reorder
153 subu $sp,$sp,48
154 .cpstore 16

```

```

155 $LCFI5:
156     sw    $ra,40($sp)
157 $LCFI6:
158     sw    $fp,36($sp)
159 $LCFI7:
160     sw    $gp,32($sp)
161 $LCFI8:
162     move   $fp,$sp
163 $LCFI9:
164     sw    $a0,48($fp)
165     sw    $a1,52($fp)
166     sw    $a2,56($fp)
167     sw    $a3,60($fp)
168     .loc 1 30 0
169 $LBB3:
170 $LBB4:
171     sw    $zero,24($fp)
172 $L19:
173     lw    $v0,24($fp)
174     lw    $v1,56($fp)
175     slt    $v0,$v0,$v1
176     bne    $v0,$zero,$L22
177     b     $L20
178 $L22:
179     .loc 1 31 0
180 $LBB5:
181     sw    $zero,28($fp)
182 $L23:
183     lw    $v0,28($fp)
184     lw    $v1,60($fp)
185     slt    $v0,$v0,$v1
186     bne    $v0,$zero,$L26
187     b     $L21
188 $L26:
189     .loc 1 32 0
190     lw    $v0,24($fp)
191     sll    $v1,$v0,2
192     lw    $v0,52($fp)
193     addu   $a0,$v1,$v0
194     lw    $v0,28($fp)
195     sll    $v1,$v0,2
196     lw    $v0,0($a0)
197     addu   $v0,$v1,$v0
198     lw    $a0,48($fp)
199     la     $a1,$LC6
200     lw    $a2,0($v0)
201     la     $t9,fprintf
202     jal    $ra,$t9
203     .loc 1 31 0
204     lw    $v0,28($fp)

```



```

205    addu    $v0,$v0,1
206    sw     $v0,28($fp)
207    b      $L23
208    .loc   1 30 0
209 $L21:
210 $LBE5:
211    lw     $v0,24($fp)
212    addu    $v0,$v0,1
213    sw     $v0,24($fp)
214    b      $L19
215 $L20:
216    .loc   1 35 0
217 $LBE4:
218    lw     $a0,48($fp)
219    la     $a1,$LC7
220    la     $t9,fprintf
221    jal    $ra,$t9
222    .loc   1 36 0
223    move    $sp,$fp
224    lw     $ra,40($sp)
225    lw     $fp,36($sp)
226    addu    $sp,$sp,48
227    j      $ra
228 $LBE3:
229    .end    armar_imagenPGM
230 $LFE31:
231    .size   armar_imagenPGM,.-armar_imagenPGM
232    .rdata
233    .align  2
234 $LC8:
235    .ascii  "%f %fi \n\000"
236    .text
237    .align  2
238    .globl  imprimir_complejo
239 $LFB33:
240    .loc   1 37 0
241    .ent    imprimir_complejo
242 imprimir_complejo:
243    .frame  $fp,48,$ra    # vars= 0, regs= 3/0, args= 24,
                          extra= 8
244    .mask  0xd0000000,-8
245    .fmask 0x00000000,0
246    .set   noreorder
247    .cplod $t9
248    .set   reorder
249    subu   $sp,$sp,48
250    .cpstore 24
251 $LCFI10:
252    sw     $ra,40($sp)
253 $LCFI11:

```

```

254  sw    $fp,36($sp)
255 $LCFI12:
256  sw    $gp,32($sp)
257 $LCFI13:
258  move  $fp,$sp
259 $LCFI14:
260  sw    $a0,48($fp)
261  sw    $a1,52($fp)
262  sw    $a2,56($fp)
263  sw    $a3,60($fp)
264  .loc 1 38 0
265  l.s   $f0,56($fp)
266  cvt.d.s $f0,$f0
267  s.d   $f0,16($sp)
268  la    $a0,$LC8
269  lw    $a2,48($fp)
270  lw    $a3,52($fp)
271  la    $t9,printf
272  jal   $ra,$t9
273  .loc 1 39 0
274  move  $sp,$fp
275  lw    $ra,40($sp)
276  lw    $fp,36($sp)
277  addu  $sp,$sp,48
278  j     $ra
279  .end  imprimir_complejo
280 $LFE33:
281  .size  imprimir_complejo , .-imprimir_complejo
282  .rdata
283  .align 3
284 $LC9:
285  .word 0
286  .word 1073741824
287  .text
288  .align 2
289  .globl abs_cplx
290 $LFB35:
291  .loc 1 41 0
292  .ent  abs_cplx
293 abs_cplx:
294  .frame $fp,48,$ra    # vars= 0, regs= 3/1, args= 16,
                        extra= 8
295  .mask 0xd0000000,-16
296  .fmask 0x00300000,-8
297  .set  noreorder
298  .cpld $t9
299  .set  reorder
300  subu  $sp,$sp,48
301  .cprestore 16
302 $LCFI15:

```

```

303  sw  $ra,32($sp)
304 $LCFI16:
305  sw  $fp,28($sp)
306 $LCFI17:
307  sw  $gp,24($sp)
308 $LCFI18:
309  s.d  $f20,40($sp)
310 $LCFI19:
311  move $fp,$sp
312 $LCFI20:
313  sw  $a0,48($fp)
314  sw  $a1,52($fp)
315  sw  $a2,56($fp)
316  sw  $a3,60($fp)
317  .loc 1 42 0
318  l.d  $f0,$LC9
319  l.d  $f12,48($fp)
320  mov.d $f14,$f0
321  la   $t9,pow
322  jal  $ra,$t9
323  mov.d $f20,$f0
324  l.s  $f0,56($fp)
325  cvt.d.s $f0,$f0
326  l.d  $f2,$LC9
327  mov.d $f12,$f0
328  mov.d $f14,$f2
329  la   $t9,pow
330  jal  $ra,$t9
331  add.d $f0,$f20,$f0
332  mov.d $f12,$f0
333  la   $t9,sqrt
334  jal  $ra,$t9
335  .loc 1 43 0
336  move $sp,$fp
337  lw   $ra,32($sp)
338  lw   $fp,28($sp)
339  l.d  $f20,40($sp)
340  addu $sp,$sp,48
341  j    $ra
342  .end  abs_cplx
343 $LFE35:
344  .size abs_cplx,.-abs_cplx
345  .rdata
346  .align 3
347 $LC10:
348  .word 0
349  .word 1073741824
350  .text
351  .align 2
352  .globl sqr_cplx

```

```

353 $LFB37:
354     .loc 1 45 0
355     .ent  sqr_cplx
356 sqr_cplx:
357     .frame $fp,56,$ra    # vars= 8, regs= 4/1, args= 16,
                          extra= 8
358     .mask 0xd0010000,-12
359     .fmask 0x00300000,-8
360     .set  noreorder
361     .cpld $t9
362     .set  reorder
363     subu  $sp,$sp,56
364     .cpstore 16
365 $LCFI21:
366     sw    $ra,44($sp)
367 $LCFI22:
368     sw    $fp,40($sp)
369 $LCFI23:
370     sw    $gp,36($sp)
371 $LCFI24:
372     sw    $s0,32($sp)
373 $LCFI25:
374     s.d   $f20,48($sp)
375 $LCFI26:
376     move  $fp,$sp
377 $LCFI27:
378     sw    $a0,56($fp)
379     .loc 1 46 0
380 $LBB6:
381     lw    $v0,56($fp)
382     l.s   $f0,8($v0)
383     cvt.d.s $f0,$f0
384     s.d   $f0,24($fp)
385     .loc 1 47 0
386     lw    $v1,56($fp)
387     lw    $v0,56($fp)
388     l.d   $f0,0($v0)
389     add.d $f2,$f0,$f0
390     lw    $v0,56($fp)
391     l.s   $f0,8($v0)
392     cvt.d.s $f0,$f0
393     mul.d $f0,$f2,$f0
394     cvt.s.d $f0,$f0
395     s.s   $f0,8($v1)
396     .loc 1 48 0
397     lw    $s0,56($fp)
398     lw    $v0,56($fp)
399     l.d   $f0,$LC10
400     l.d   $f12,0($v0)
401     mov.d $f14,$f0

```

```

402  la    $t9,pow
403  jal   $ra,$t9
404  mov.d $f20,$f0
405  l.d   $f0,$LC10
406  l.d   $f12,24($fp)
407  mov.d $f14,$f0
408  la    $t9,pow
409  jal   $ra,$t9
410  sub.d $f0,$f20,$f0
411  s.d   $f0,0($s0)
412  .loc  1 49 0
413  move  $sp,$fp
414  lw    $ra,44($sp)
415  lw    $fp,40($sp)
416  lw    $s0,32($sp)
417  l.d   $f20,48($sp)
418  addu  $sp,$sp,56
419  j     $ra
420 $LBE6:
421  .end   sqr_cplx
422 $LFE37:
423  .size  sqr_cplx,.-sqr_cplx
424  .rdata
425  .align 2
426 $LC11:
427  .ascii "Debe ingresar correctamente los argumentos.
        Abortando ej"
428  .ascii "ecucion\n\000"
429  .align 2
430 $LC12:
431  .ascii "La ruta de archivo ingresada no es valida.
        Abortando ejec"
432  .ascii "ucion\n\000"
433  .align 2
434 $LC13:
435  .ascii "No se ha podido reservar la memoria
        necesaria. Abortando"
436  .ascii " ejecucion\n\000"
437  .align 2
438 $LC14:
439  .ascii "Error no contemplado. Abortando Ejecucion
        \n\000"
440  .text
441  .align 2
442  .globl imprimir_error
443 $LFB39:
444  .loc  1 51 0
445  .ent   imprimir_error
446 imprimir_error:

```

```

447 .frame $fp,48,$ra    # vars= 8, regs= 3/0, args= 16,
    extra= 8
448 .mask 0xd0000000,-8
449 .fmask 0x00000000,0
450 .set noreorder
451 .cpload $t9
452 .set reorder
453 subu $sp,$sp,48
454 .cpstore 16
455 $LCFI28:
456 sw $ra,40($sp)
457 $LCFI29:
458 sw $fp,36($sp)
459 $LCFI30:
460 sw $gp,32($sp)
461 $LCFI31:
462 move $fp,$sp
463 $LCFI32:
464 sw $a0,48($fp)
465 .loc 1 52 0
466 lw $v0,48($fp)
467 sw $v0,24($fp)
468 li $v0,-2          # 0xfffffffffffffffe
469 lw $v1,24($fp)
470 beq $v1,$v0,$L33
471 lw $v1,24($fp)
472 slt $v0,$v1,-1
473 beq $v0,$zero,$L37
474 li $v0,-3          # 0xfffffffffffffffd
475 lw $v1,24($fp)
476 beq $v1,$v0,$L34
477 b $L35
478 $L37:
479 li $v0,-1          # 0xffffffffffffffff
480 lw $v1,24($fp)
481 beq $v1,$v0,$L32
482 b $L35
483 $L32:
484 .loc 1 54 0
485 la $a0,$LC11
486 la $t9,printf
487 jal $ra,$t9
488 .loc 1 55 0
489 li $a0,-1          # 0xffffffffffffffff
490 la $t9,exit
491 jal $ra,$t9
492 $L33:
493 .loc 1 58 0
494 la $a0,$LC12
495 la $t9,printf

```

```

496     jal $ra,$t9
497     .loc 1 59 0
498     li $a0,-2          # 0xfffffffffffffffe
499     la $t9,exit
500     jal $ra,$t9
501 $L34:
502     .loc 1 61 0
503     la $a0,$LC13
504     la $t9,printf
505     jal $ra,$t9
506     .loc 1 62 0
507     li $a0,-3          # 0xfffffffffffffffd
508     la $t9,exit
509     jal $ra,$t9
510 $L35:
511     .loc 1 64 0
512     la $a0,$LC14
513     la $t9,printf
514     jal $ra,$t9
515     .loc 1 65 0
516     lw $a0,48($fp)
517     la $t9,exit
518     jal $ra,$t9
519     .loc 1 66 0
520     .end imprimir_error
521 $LFE39:
522     .size imprimir_error,.-imprimir_error
523     .align 2
524     .globl create_matrix
525 $LFB41:
526     .loc 1 68 0
527     .ent create_matrix
528 create_matrix:
529     .frame $fp,48,$ra    # vars= 8, regs= 4/0, args= 16,
                          extra= 8
530     .mask 0xd0010000,-4
531     .fmask 0x00000000,0
532     .set noreorder
533     .cpload $t9
534     .set reorder
535     subu $sp,$sp,48
536     .cpstore 16
537 $LCFI33:
538     sw $ra,44($sp)
539 $LCFI34:
540     sw $fp,40($sp)
541 $LCFI35:
542     sw $gp,36($sp)
543 $LCFI36:
544     sw $s0,32($sp)

```

```

545 $LCFI37:
546     move    $fp,$sp
547 $LCFI38:
548     sw      $a0,48($fp)
549     sw      $a1,52($fp)
550     sw      $a2,56($fp)
551     .loc    1 69 0
552 $LBB7:
553     lw      $v0,48($fp)
554     sll     $v0,$v0,2
555     move    $a0,$v0
556     la      $t9, malloc
557     jal     $ra,$t9
558     sw      $v0,24($fp)
559     .loc    1 71 0
560     lw      $v0,24($fp)
561     bne     $v0,$zero,$L39
562     .loc    1 72 0
563     li      $v0,-3          # 0xffffffffffffffffffd
564     sw      $v0,56($fp)
565     .loc    1 73 0
566     lw      $a0,56($fp)
567     la      $t9, imprimir_error
568     jal     $ra,$t9
569 $L39:
570     .loc    1 75 0
571 $LBB8:
572     sw      $zero,28($fp)
573 $L40:
574     lw      $v0,28($fp)
575     lw      $v1,48($fp)
576     slt     $v0,$v0,$v1
577     bne     $v0,$zero,$L43
578     b       $L41
579 $L43:
580     .loc    1 76 0
581     lw      $v0,28($fp)
582     sll     $v1,$v0,2
583     lw      $v0,24($fp)
584     addu    $s0,$v1,$v0
585     lw      $v0,52($fp)
586     sll     $v0,$v0,2
587     move    $a0,$v0
588     la      $t9, malloc
589     jal     $ra,$t9
590     sw      $v0,0($s0)
591     .loc    1 77 0
592     lw      $v0,28($fp)
593     sll     $v1,$v0,2
594     lw      $v0,24($fp)

```



```

595  addu  $v0,$v1,$v0
596  lw    $v0,0($v0)
597  bne   $v0,$zero,$L42
598  .loc  1 78 0
599  li    $v0,-3      # 0xfffffffffffffffd
600  sw    $v0,56($fp)
601  .loc  1 79 0
602  lw    $a0,56($fp)
603  la    $t9,imprimir_error
604  jal   $ra,$t9
605  .loc  1 75 0
606 $L42:
607  lw    $v0,28($fp)
608  addu  $v0,$v0,1
609  sw    $v0,28($fp)
610  b     $L40
611 $L41:
612  .loc  1 82 0
613 $LBE8:
614  lw    $v0,24($fp)
615  .loc  1 83 0
616  move  $sp,$fp
617  lw    $ra,44($sp)
618  lw    $fp,40($sp)
619  lw    $s0,32($sp)
620  addu  $sp,$sp,48
621  j     $ra
622 $LBE7:
623 .end   create_matrix
624 $LFE41:
625 .size  create_matrix,.-create_matrix
626 .rdata
627 .align 3
628 $LC15:
629 .word 0
630 .word 1073741824
631 .text
632 .align 2
633 .globl generate_julia
634 $LFB43:
635 .loc 1 85 0
636 .ent   generate_julia
637 generate_julia:
638 .frame $fp,136,$ra  # vars= 96, regs= 3/0, args= 16,
        extra= 8
639 .mask 0xd0000000,-8
640 .fmask 0x00000000,0
641 .set   noreorder
642 .cplod $t9
643 .set   reorder

```

```

644    subu    $sp,$sp,136
645    .cprestore 16
646 $LCFI39:
647    sw      $ra,128($sp)
648 $LCFI40:
649    sw      $fp,124($sp)
650 $LCFI41:
651    sw      $gp,120($sp)
652 $LCFI42:
653    move    $fp,$sp
654 $LCFI43:
655    sw      $a0,136($fp)
656    sw      $a1,140($fp)
657    sw      $a2,144($fp)
658    .loc 1 87 0
659 $LBB9:
660    sw      $zero,32($fp)
661    .loc 1 88 0
662    l.d     $f0,152($fp)
663    neg.d   $f2,$f0
664    l.d     $f0,$LC15
665    div.d   $f0,$f2,$f0
666    s.d     $f0,40($fp)
667    .loc 1 89 0
668    l.d     $f2,152($fp)
669    l.d     $f0,$LC15
670    div.d   $f0,$f2,$f0
671    s.d     $f0,48($fp)
672    .loc 1 90 0
673    l.d     $f0,160($fp)
674    neg.d   $f2,$f0
675    l.d     $f0,$LC15
676    div.d   $f0,$f2,$f0
677    s.d     $f0,56($fp)
678    .loc 1 91 0
679    l.d     $f2,160($fp)
680    l.d     $f0,$LC15
681    div.d   $f0,$f2,$f0
682    s.d     $f0,64($fp)
683    .loc 1 93 0
684    lw      $v1,140($fp)
685    li      $v0,1      # 0x1
686    beq     $v1,$v0,$L46
687    .loc 1 94 0
688    l.d     $f2,48($fp)
689    l.d     $f0,40($fp)
690    sub.d   $f2,$f2,$f0
691    lw      $v0,140($fp)
692    addu    $v0,$v0,-1
693    mtc1    $v0,$f0

```

```

694    cvt.d.w $f0,$f0
695    div.d $f0,$f2,$f0
696    s.d $f0,72($fp)
697    b $L47
698 $L46:
699    .loc 1 97 0
700    l.d $f0,48($fp)
701    l.d $f2,40($fp)
702    sub.d $f0,$f0,$f2
703    s.d $f0,72($fp)
704 $L47:
705    .loc 1 99 0
706    lw $v1,144($fp)
707    li $v0,1      # 0x1
708    beq $v1,$v0,$L48
709    .loc 1 100 0
710    l.d $f2,64($fp)
711    l.d $f0,56($fp)
712    sub.d $f2,$f2,$f0
713    lw $v0,144($fp)
714    addu $v0,$v0,-1
715    mtc1 $v0,$f0
716    cvt.d.w $f0,$f0
717    div.d $f0,$f2,$f0
718    s.d $f0,80($fp)
719    b $L49
720 $L48:
721    .loc 1 103 0
722    l.d $f0,64($fp)
723    l.d $f2,56($fp)
724    sub.d $f0,$f0,$f2
725    s.d $f0,80($fp)
726 $L49:
727    .loc 1 108 0
728 $LBB10:
729    sw $zero,88($fp)
730 $L50:
731    lw $v0,88($fp)
732    lw $v1,144($fp)
733    slt $v0,$v0,$v1
734    bne $v0,$zero,$L53
735    b $L45
736 $L53:
737    .loc 1 109 0
738 $LBB11:
739    l.d $f2,80($fp)
740    l.d $f0,$LC15
741    div.d $f2,$f2,$f0
742    l.d $f0,64($fp)
743    sub.d $f4,$f0,$f2

```

```

744    l.s  $f0,88($fp)
745    cvt.d.w $f2,$f0
746    l.d  $f0,80($fp)
747    mul.d $f0,$f2,$f0
748    sub.d $f2,$f4,$f0
749    l.s  $f0,192($fp)
750    cvt.d.s $f0,$f0
751    add.d $f0,$f2,$f0
752    s.d  $f0,24($fp)
753    .loc 1 111 0
754 $LBB12:
755    sw   $zero,112($fp)
756 $L54:
757    lw   $v0,112($fp)
758    lw   $v1,140($fp)
759    slt  $v0,$v0,$v1
760    bne  $v0,$zero,$L57
761    b    $L52
762 $L57:
763    .loc 1 112 0
764    l.d  $f2,72($fp)
765    l.d  $f0,$LC15
766    div.d $f2,$f2,$f0
767    l.d  $f0,40($fp)
768    add.d $f4,$f2,$f0
769    l.s  $f0,112($fp)
770    cvt.d.w $f2,$f0
771    l.d  $f0,72($fp)
772    mul.d $f0,$f2,$f0
773    add.d $f2,$f4,$f0
774    l.d  $f0,184($fp)
775    add.d $f0,$f2,$f0
776    s.d  $f0,96($fp)
777    .loc 1 113 0
778    l.d  $f0,24($fp)
779    cvt.s.d $f0,$f0
780    s.s  $f0,104($fp)
781    .loc 1 115 0
782 $L58:
783    lw   $a0,96($fp)
784    lw   $a1,100($fp)
785    lw   $a2,104($fp)
786    lw   $a3,108($fp)
787    la   $t9,abs_cplx
788    jal  $ra,$t9
789    mov.d $f2,$f0
790    l.d  $f0,$LC15
791    c.lt.d $f2,$f0
792    bc1t $L62
793    b    $L59

```

```

794 $L62:
795     lw    $v0,32($fp)
796     slt   $v0,$v0,255
797     bne   $v0,$zero,$L60
798     b     $L59
799 $L60:
800     .loc 1 117 0
801     addu   $v0,$fp,96
802     move   $a0,$v0
803     la     $t9,sqr_cplx
804     jal    $ra,$t9
805     .loc 1 118 0
806     l.d    $f2,96($fp)
807     l.d    $f0,168($fp)
808     add.d   $f0,$f2,$f0
809     s.d    $f0,96($fp)
810     .loc 1 119 0
811     l.s    $f2,104($fp)
812     l.s    $f0,176($fp)
813     add.s   $f0,$f2,$f0
814     s.s    $f0,104($fp)
815     .loc 1 120 0
816     lw     $v0,32($fp)
817     addu   $v0,$v0,1
818     sw     $v0,32($fp)
819     b     $L58
820 $L59:
821     .loc 1 122 0
822     lw     $v0,88($fp)
823     sll    $v1,$v0,2
824     lw     $v0,136($fp)
825     addu   $a0,$v1,$v0
826     lw     $v0,112($fp)
827     sll    $v1,$v0,2
828     lw     $v0,0($a0)
829     addu   $v1,$v1,$v0
830     lw     $v0,32($fp)
831     sw     $v0,0($v1)
832     .loc 1 123 0
833     sw     $zero,32($fp)
834     .loc 1 111 0
835     lw     $v0,112($fp)
836     addu   $v0,$v0,1
837     sw     $v0,112($fp)
838     b     $L54
839     .loc 1 108 0
840 $L52:
841 $LBE12:
842 $LBE11:
843     lw     $v0,88($fp)

```

```

844    addu    $v0,$v0,1
845    sw      $v0,88($fp)
846    b       $L50
847    .loc    1 126 0
848 $L45:
849    move    $sp,$fp
850    lw      $ra,128($sp)
851    lw      $fp,124($sp)
852    addu    $sp,$sp,136
853    j       $ra
854 $LBE10:
855 $LBE9:
856    .end    generate_julia
857 $LFE43:
858    .size   generate_julia , .-generate_julia
859    .rdata
860    .align   2
861 $LC19:
862    .ascii  "Usage:\n"
863    .ascii  "    tp0 -h  -V -c <a+bi> -C <a+bi> -H <float>
            -w <float> -"
864    .ascii  "o <out_file> -\n"
865    .ascii  "Options:\n"
866    .ascii  "    -V\t    Imprime la version y finaliza.\n"
867    .ascii  "    -h\t    Imprime esta informacion y
            finaliza.\n"
868    .ascii  "    -c\t    Setea el centro de la imagen.\n"
869    .ascii  "    -H\t    Setea el alto del rectangulo.
            Valor por defe"
870    .ascii  "cto=4\n"
871    .ascii  "    -w\t    Setea el ancho del rectangulo.
            Valor por def"
872    .ascii  "ecto=4\n"
873    .ascii  "    -o\t    Setea el archivo de salida
            -C\t    Setea "
874    .ascii  "la constante del algoritmo. Valor por
            defecto= 0.285+0.0"
875    .ascii  "li\n"
876    .ascii  "Examples:\n"
877    .ascii  "    tp0 -c +0.282-0.01i -w 0.005 -H 0.005 -o
            dos.pgm\n\000"
878    .align   2
879 $LC20:
880    .ascii  "Conjunto de Julia\n"
881    .ascii  "v1.0\n\000"
882    .align   2
883 $LC21:
884    .ascii  "xX\000"
885    .align   2
886 $LC22:

```

```

887 .ascii "+\000"
888 .align 2
889 $LC23:
890 .ascii "i\000"
891 .align 2
892 $LC24:
893 .ascii "-\000"
894 .align 2
895 $LC25:
896 .ascii "w\000"
897 .align 2
898 $LC26:
899 .ascii "Argumento desconocido: prueba con -h para ver
      la ayuda.\n"
900 .ascii "\000"
901 .align 2
902 $LC27:
903 .ascii "Error! Formato desconocido. Prueba con -h
      para ver la ay"
904 .ascii "uda. \n\000"
905 .align 2
906 $LC28:
907 .ascii "Se corra el programa con los valores por
      DEFAULT. \n\000"
908 .align 3
909 $LC16:
910 .word -1546188227
911 .word 1070742896
912 .align 2
913 $LC17:
914 .word -1138501878
915 .align 3
916 $LC18:
917 .word 0
918 .word 1074790400
919 .text
920 .align 2
921 .globl main
922 $LFB45:
923 .loc 1 129 0
924 .ent main
925 main:
926 .frame $fp,176,$ra # vars= 88, regs= 3/0, args= 64,
      extra= 8
927 .mask 0xd0000000,-8
928 .fmask 0x00000000,0
929 .set noreorder
930 .cpld $t9
931 .set reorder
932 subu $sp,$sp,176

```

```

933     .cprestore 64
934 $LCFI44:
935     sw     $ra,168($sp)
936 $LCFI45:
937     sw     $fp,164($sp)
938 $LCFI46:
939     sw     $gp,160($sp)
940 $LCFI47:
941     move   $fp,$sp
942 $LCFI48:
943     sw     $a0,176($fp)
944     sw     $a1,180($fp)
945     .loc 1 130 0
946 $LBB13:
947     sw     $zero,72($fp)
948     .loc 1 131 0
949     li     $v0,480      # 0x1e0
950     sw     $v0,76($fp)
951     .loc 1 132 0
952     li     $v0,640      # 0x280
953     sw     $v0,80($fp)
954     .loc 1 135 0
955     sw     $zero,104($fp)
956     sw     $zero,108($fp)
957     .loc 1 136 0
958     sw     $zero,112($fp)
959     .loc 1 137 0
960     l.d    $f0,$LC16
961     s.d    $f0,88($fp)
962     .loc 1 138 0
963     l.s    $f0,$LC17
964     s.s    $f0,96($fp)
965     .loc 1 141 0
966     l.d    $f0,$LC18
967     s.d    $f0,128($fp)
968     l.d    $f0,$LC18
969     s.d    $f0,136($fp)
970     .loc 1 142 0
971     la     $v0,--sF+88
972     sw     $v0,144($fp)
973     .loc 1 144 0
974     lw     $v0,176($fp)
975     slt    $v0,$v0,2
976     bne    $v0,$zero,$L64
977     .loc 1 145 0
978 $LBB14:
979     li     $v0,1        # 0x1
980     sw     $v0,148($fp)
981 $L65:
982     lw     $v0,148($fp)

```



```

983    lw    $v1,176($fp)
984    slt   $v0,$v0,$v1
985    bne   $v0,$zero,$L68
986    b     $L101
987 $L68:
988    .loc 1 146 0
989    lw    $v0,148($fp)
990    sll   $v1,$v0,2
991    lw    $v0,180($fp)
992    addu   $v0,$v1,$v0
993    lw    $v0,0($v0)
994    lb    $v1,0($v0)
995    li    $v0,45          # 0x2d
996    bne   $v1,$v0,$L69
997    .loc 1 148 0
998    lw    $v0,148($fp)
999    sll   $v1,$v0,2
1000    lw    $v0,180($fp)
1001    addu   $v0,$v1,$v0
1002    lw    $v0,0($v0)
1003    addu   $v0,$v0,1
1004    lb    $v0,0($v0)
1005    addu   $v0,$v0,-67
1006    sw    $v0,156($fp)
1007    lw    $v1,156($fp)
1008    sltu   $v0,$v1,53
1009    beq    $v0,$zero,$L98
1010    lw    $v0,156($fp)
1011    sll   $v1,$v0,2
1012    la     $v0,$L99
1013    addu   $v0,$v1,$v0
1014    lw    $v0,0($v0)
1015    .cpadd $v0
1016    j      $v0
1017    .rdata
1018    .align 2
1019 $L99:
1020    .gpword $L78
1021    .gpword $L98
1022    .gpword $L98
1023    .gpword $L98
1024    .gpword $L98
1025    .gpword $L90
1026    .gpword $L98
1027    .gpword $L98
1028    .gpword $L98
1029    .gpword $L98
1030    .gpword $L98
1031    .gpword $L98
1032    .gpword $L98

```

```

1033 .gpword $L98
1034 .gpword $L98
1035 .gpword $L98
1036 .gpword $L98
1037 .gpword $L98
1038 .gpword $L98
1039 .gpword $L72
1040 .gpword $L98
1041 .gpword $L98
1042 .gpword $L98
1043 .gpword $L98
1044 .gpword $L98
1045 .gpword $L98
1046 .gpword $L98
1047 .gpword $L98
1048 .gpword $L98
1049 .gpword $L98
1050 .gpword $L98
1051 .gpword $L98
1052 .gpword $L84
1053 .gpword $L98
1054 .gpword $L98
1055 .gpword $L98
1056 .gpword $L98
1057 .gpword $L71
1058 .gpword $L98
1059 .gpword $L98
1060 .gpword $L98
1061 .gpword $L98
1062 .gpword $L98
1063 .gpword $L98
1064 .gpword $L92
1065 .gpword $L98
1066 .gpword $L98
1067 .gpword $L73
1068 .gpword $L98
1069 .gpword $L98
1070 .gpword $L98
1071 .gpword $L98
1072 .gpword $L96
1073 .text
1074 $L71:
1075 .loc 1 149 0
1076 la $a0,$LC19
1077 la $t9,printf
1078 jal $ra,$t9
1079 .loc 1 160 0
1080 sw $zero,152($fp)
1081 b $L63
1082 $L72:

```

```

1083 .loc 1 162 0
1084 la $a0,$LC20
1085 la $t9,printf
1086 jal $ra,$t9
1087 .loc 1 163 0
1088 sw $zero,152($fp)
1089 b $L63
1090 $L73:
1091 .loc 1 166 0
1092 lw $v0,148($fp)
1093 sll $v1,$v0,2
1094 lw $v0,180($fp)
1095 addu $v0,$v1,$v0
1096 addu $v0,$v0,4
1097 lw $a0,0($v0)
1098 la $a1,$LC21
1099 la $t9,strtok
1100 jal $ra,$t9
1101 sw $v0,120($fp)
1102 .loc 1 167 0
1103 lw $v0,120($fp)
1104 bne $v0,$zero,$L74
1105 .loc 1 168 0
1106 li $v0,-1 # 0xffffffffffffffff
1107 sw $v0,72($fp)
1108 .loc 1 169 0
1109 b $L67
1110 $L74:
1111 .loc 1 171 0
1112 lw $a0,120($fp)
1113 la $t9,atoi
1114 jal $ra,$t9
1115 sw $v0,80($fp)
1116 .loc 1 173 0
1117 move $a0,$zero
1118 la $a1,$LC4
1119 la $t9,strtok
1120 jal $ra,$t9
1121 sw $v0,120($fp)
1122 .loc 1 174 0
1123 lw $v0,120($fp)
1124 bne $v0,$zero,$L75
1125 .loc 1 175 0
1126 li $v0,-1 # 0xffffffffffffffff
1127 sw $v0,72($fp)
1128 .loc 1 176 0
1129 b $L67
1130 $L75:
1131 .loc 1 178 0
1132 lw $a0,120($fp)

```

```

1133  la  $t9,atoi
1134  jal $ra,$t9
1135  sw  $v0,76($fp)
1136  .loc 1 179 0
1137  lw  $v0,80($fp)
1138  beq $v0,$zero,$L77
1139  lw  $v0,76($fp)
1140  bne $v0,$zero,$L67
1141 $L77:
1142  .loc 1 180 0
1143  li  $v0,-1      # 0xffffffffffffffff
1144  sw  $v0,72($fp)
1145  .loc 1 181 0
1146  b  $L67
1147 $L78:
1148  .loc 1 185 0
1149  lw  $v0,148($fp)
1150  sll $v1,$v0,2
1151  lw  $v0,180($fp)
1152  addu $v0,$v1,$v0
1153  addu $v0,$v0,4
1154  lw  $v0,0($v0)
1155  sw  $v0,120($fp)
1156  .loc 1 186 0
1157  lw  $v0,120($fp)
1158  bne $v0,$zero,$L79
1159  .loc 1 187 0
1160  li  $v0,-1      # 0xffffffffffffffff
1161  sw  $v0,72($fp)
1162  .loc 1 188 0
1163  b  $L67
1164 $L79:
1165  .loc 1 190 0
1166  lw  $a0,120($fp)
1167  la  $t9,atof
1168  jal $ra,$t9
1169  s.d $f0,88($fp)
1170  .loc 1 191 0
1171  lw  $v0,148($fp)
1172  sll $v1,$v0,2
1173  lw  $v0,180($fp)
1174  addu $v0,$v1,$v0
1175  addu $v0,$v0,4
1176  lw  $v0,0($v0)
1177  lb  $v1,0($v0)
1178  li  $v0,45      # 0x2d
1179  beq $v1,$v0,$L81
1180  lw  $v0,148($fp)
1181  sll $v1,$v0,2
1182  lw  $v0,180($fp)

```

```

1183    addu    $v0,$v1,$v0
1184    addu    $v0,$v0,4
1185    lw      $v0,0($v0)
1186    lb      $v1,0($v0)
1187    li      $v0,43          # 0x2b
1188    beq     $v1,$v0,$L81
1189    b       $L80
1190 $L81:
1191    .loc 1 192 0
1192    lw      $v0,120($fp)
1193    addu    $v0,$v0,1
1194    move    $a0,$v0
1195    la      $a1,$LC22
1196    la      $t9, strpbrk
1197    jal     $ra,$t9
1198    sw      $v0,120($fp)
1199    b       $L82
1200 $L80:
1201    .loc 1 194 0
1202    lw      $a0,120($fp)
1203    la      $a1,$LC22
1204    la      $t9, strpbrk
1205    jal     $ra,$t9
1206    sw      $v0,120($fp)
1207 $L82:
1208    .loc 1 195 0
1209    lw      $a0,120($fp)
1210    la      $a1,$LC23
1211    la      $t9, strpbrk
1212    jal     $ra,$t9
1213    sw      $v0,124($fp)
1214    .loc 1 196 0
1215    lw      $v0,124($fp)
1216    bne     $v0,$zero,$L83
1217    .loc 1 197 0
1218    li      $v0,-1          # 0xffffffffffffffff
1219    sw      $v0,72($fp)
1220    .loc 1 198 0
1221    b       $L67
1222 $L83:
1223    .loc 1 200 0
1224    lw      $a0,120($fp)
1225    la      $t9, atof
1226    jal     $ra,$t9
1227    cvt.s.d $f0,$f0
1228    s.s     $f0,96($fp)
1229    .loc 1 201 0
1230    b       $L67
1231 $L84:
1232    .loc 1 204 0

```

```

1233 lw $v0,148($fp)
1234 sll $v1,$v0,2
1235 lw $v0,180($fp)
1236 addu $v0,$v1,$v0
1237 addu $v0,$v0,4
1238 lw $v0,0($v0)
1239 sw $v0,120($fp)
1240 .loc 1 205 0
1241 lw $v0,120($fp)
1242 bne $v0,$zero,$L85
1243 .loc 1 206 0
1244 li $v0,-1 # 0xffffffffffffffff
1245 sw $v0,72($fp)
1246 .loc 1 207 0
1247 b $L67
1248 $L85:
1249 .loc 1 209 0
1250 lw $a0,120($fp)
1251 la $t9,atof
1252 jal $ra,$t9
1253 s.d $f0,104($fp)
1254 .loc 1 210 0
1255 lw $v0,148($fp)
1256 sll $v1,$v0,2
1257 lw $v0,180($fp)
1258 addu $v0,$v1,$v0
1259 addu $v0,$v0,4
1260 lw $v0,0($v0)
1261 lb $v1,0($v0)
1262 li $v0,45 # 0x2d
1263 beq $v1,$v0,$L87
1264 lw $v0,148($fp)
1265 sll $v1,$v0,2
1266 lw $v0,180($fp)
1267 addu $v0,$v1,$v0
1268 addu $v0,$v0,4
1269 lw $v0,0($v0)
1270 lb $v1,0($v0)
1271 li $v0,43 # 0x2b
1272 beq $v1,$v0,$L87
1273 b $L86
1274 $L87:
1275 .loc 1 211 0
1276 lw $v0,120($fp)
1277 addu $v0,$v0,1
1278 move $a0,$v0
1279 la $a1,$LC22
1280 la $t9,strupbrk
1281 jal $ra,$t9
1282 sw $v0,120($fp)

```

```

1283    b $L88
1284 $L86:
1285    .loc 1 214 0
1286    lw  $a0,120($fp)
1287    la  $a1,$LC22
1288    la  $t9, strpbrk
1289    jal $ra,$t9
1290    sw  $v0,120($fp)
1291 $L88:
1292    .loc 1 216 0
1293    lw  $a0,120($fp)
1294    la  $a1,$LC23
1295    la  $t9, strpbrk
1296    jal $ra,$t9
1297    sw  $v0,124($fp)
1298    .loc 1 217 0
1299    lw  $v0,124($fp)
1300    bne $v0,$zero,$L89
1301    .loc 1 218 0
1302    li  $v0,-1          # 0xffffffffffffffff
1303    sw  $v0,72($fp)
1304    .loc 1 219 0
1305    b $L67
1306 $L89:
1307    .loc 1 221 0
1308    lw  $a0,120($fp)
1309    la  $t9, atof
1310    jal $ra,$t9
1311    cvt.s.d $f0,$f0
1312    s.s $f0,112($fp)
1313    .loc 1 222 0
1314    b $L67
1315 $L90:
1316    .loc 1 224 0
1317    lw  $v0,148($fp)
1318    sll $v1,$v0,2
1319    lw  $v0,180($fp)
1320    addu $v0,$v1,$v0
1321    addu $v0,$v0,4
1322    lw  $v0,0($v0)
1323    bne $v0,$zero,$L91
1324    .loc 1 225 0
1325    li  $v0,-1          # 0xffffffffffffffff
1326    sw  $v0,72($fp)
1327    .loc 1 226 0
1328    b $L67
1329 $L91:
1330    .loc 1 228 0
1331    lw  $v0,148($fp)
1332    sll $v1,$v0,2

```

```

1333 lw    $v0,180($fp)
1334 addu   $v0,$v1,$v0
1335 addu   $v0,$v0,4
1336 lw    $a0,0($v0)
1337 la     $t9,atof
1338 jal    $ra,$t9
1339 s.d    $f0,128($fp)
1340 .loc 1 229 0
1341 b      $L67
1342 $L92:
1343 .loc 1 231 0
1344 lw    $v0,148($fp)
1345 sll   $v1,$v0,2
1346 lw    $v0,180($fp)
1347 addu   $v0,$v1,$v0
1348 addu   $v0,$v0,4
1349 lw    $v0,0($v0)
1350 bne    $v0,$zero,$L93
1351 .loc 1 232 0
1352 li     $v0,-1          # 0xffffffffffffffff
1353 sw     $v0,72($fp)
1354 .loc 1 233 0
1355 b      $L67
1356 $L93:
1357 .loc 1 235 0
1358 lw    $v0,148($fp)
1359 sll   $v1,$v0,2
1360 lw    $v0,180($fp)
1361 addu   $v0,$v1,$v0
1362 addu   $v0,$v0,4
1363 lw    $a0,0($v0)
1364 la     $a1,$LC24
1365 la     $t9,strcmp
1366 jal    $ra,$t9
1367 bne    $v0,$zero,$L94
1368 .loc 1 236 0
1369 b      $L67
1370 $L94:
1371 .loc 1 239 0
1372 lw    $v0,148($fp)
1373 sll   $v1,$v0,2
1374 lw    $v0,180($fp)
1375 addu   $v0,$v1,$v0
1376 addu   $v0,$v0,4
1377 lw    $a0,0($v0)
1378 la     $a1,$LC25
1379 la     $t9,fopen
1380 jal    $ra,$t9
1381 sw     $v0,144($fp)
1382 .loc 1 240 0

```



```

1383    lw    $v0,144($fp)
1384    bne   $v0,$zero,$L67
1385    .loc  1 241 0
1386    li    $v0,-2          # 0xfffffffffffffffe
1387    sw    $v0,72($fp)
1388    .loc  1 242 0
1389    b     $L67
1390 $L96:
1391    .loc  1 246 0
1392    lw    $v0,148($fp)
1393    sll   $v1,$v0,2
1394    lw    $v0,180($fp)
1395    addu   $v0,$v1,$v0
1396    addu   $v0,$v0,4
1397    lw    $v0,0($v0)
1398    bne   $v0,$zero,$L97
1399    .loc  1 247 0
1400    li    $v0,-1          # 0xfffffffffffffff
1401    sw    $v0,72($fp)
1402    .loc  1 248 0
1403    b     $L67
1404 $L97:
1405    .loc  1 250 0
1406    lw    $v0,148($fp)
1407    sll   $v1,$v0,2
1408    lw    $v0,180($fp)
1409    addu   $v0,$v1,$v0
1410    addu   $v0,$v0,4
1411    lw    $a0,0($v0)
1412    la    $t9,atof
1413    jal   $ra,$t9
1414    s.d   $f0,136($fp)
1415    .loc  1 251 0
1416    b     $L67
1417 $L98:
1418    .loc  1 252 0
1419    la    $a0,$LC26
1420    la    $t9,printf
1421    jal   $ra,$t9
1422    b     $L67
1423 $L69:
1424    .loc  1 256 0
1425    la    $a0,$LC27
1426    la    $t9,printf
1427    jal   $ra,$t9
1428    .loc  1 257 0
1429    li    $v0,1          # 0x1
1430    sw    $v0,152($fp)
1431    b     $L63
1432    .loc  1 145 0

```

```

1433 $L67:
1434     lw    $v0,148($fp)
1435     addu   $v0,$v0,2
1436     sw    $v0,148($fp)
1437     b     $L65
1438 $L64:
1439     .loc 1 263 0
1440 $LBE14:
1441     la    $a0,$LC28
1442     la    $t9,printf
1443     jal   $ra,$t9
1444 $L101:
1445     .loc 1 265 0
1446     lw    $v0,72($fp)
1447     beq   $v0,$zero,$L102
1448     lw    $a0,72($fp)
1449     la    $t9,imprimir_error
1450     jal   $ra,$t9
1451 $L102:
1452     .loc 1 266 0
1453     lw    $a0,144($fp)
1454     lw    $a1,76($fp)
1455     lw    $a2,80($fp)
1456     la    $t9,armar_headerPGM
1457     jal   $ra,$t9
1458     .loc 1 269 0
1459     lw    $a0,76($fp)
1460     lw    $a1,80($fp)
1461     lw    $a2,72($fp)
1462     la    $t9,create_matrix
1463     jal   $ra,$t9
1464     sw    $v0,148($fp)
1465     .loc 1 271 0
1466     l.d   $f0,136($fp)
1467     s.d   $f0,16($sp)
1468     l.d   $f0,128($fp)
1469     s.d   $f0,24($sp)
1470     lw    $v0,88($fp)
1471     sw    $v0,32($sp)
1472     lw    $v0,92($fp)
1473     sw    $v0,36($sp)
1474     lw    $v0,96($fp)
1475     sw    $v0,40($sp)
1476     lw    $v0,100($fp)
1477     sw    $v0,44($sp)
1478     lw    $v0,104($fp)
1479     sw    $v0,48($sp)
1480     lw    $v0,108($fp)
1481     sw    $v0,52($sp)
1482     lw    $v0,112($fp)

```

```

1483  sw    $v0,56($sp)
1484  lw    $v0,116($fp)
1485  sw    $v0,60($sp)
1486  lw    $a0,148($fp)
1487  lw    $a1,80($fp)
1488  lw    $a2,76($fp)
1489  la    $t9,generate_julia
1490  jal   $ra,$t9
1491  .loc 1 272 0
1492  lw    $a0,144($fp)
1493  lw    $a1,148($fp)
1494  lw    $a2,76($fp)
1495  lw    $a3,80($fp)
1496  la    $t9,armar_imagenPGM
1497  jal   $ra,$t9
1498  .loc 1 273 0
1499  sw    $zero,152($fp)
1500  .loc 1 274 0
1501  $L63:
1502  $LBE13:
1503  lw    $v0,152($fp)
1504  move   $sp,$fp
1505  lw    $ra,168($sp)
1506  lw    $fp,164($sp)
1507  addu   $sp,$sp,176
1508  j      $ra
1509  .end   main
1510  $LFE45:
1511  .size  main,.-main
1512  .section .debug_frame,"",@progbits
1513  $Lframe0:
1514  .4byte $LECIE0-$LSCIE0
1515  $LSCIE0:
1516  .4byte 0xffffffff
1517  .byte 0x1
1518  .ascii "\000"
1519  .uleb128 0x1
1520  .sleb128 4
1521  .byte 0x40
1522  .byte 0xc
1523  .uleb128 0x1d
1524  .uleb128 0x0
1525  .align 2
1526  $LECIE0:
1527  $LSFDE0:
1528  .4byte $LEFDE0-$LASFDE0
1529  $LASFDE0:
1530  .4byte $Lframe0
1531  .4byte $LFB29
1532  .4byte $LFE29-$LFB29

```

```

1533 .byte 0x4
1534 .4 byte $LCFI0-$LFB29
1535 .byte 0xe
1536 .uleb128 0x60
1537 .byte 0x4
1538 .4 byte $LCFI3-$LCFI0
1539 .byte 0x11
1540 .uleb128 0x1c
1541 .sleb128 -4
1542 .byte 0x11
1543 .uleb128 0x1e
1544 .sleb128 -3
1545 .byte 0x11
1546 .uleb128 0x40
1547 .sleb128 -2
1548 .byte 0x4
1549 .4 byte $LCFI4-$LCFI3
1550 .byte 0xc
1551 .uleb128 0x1e
1552 .uleb128 0x60
1553 .align 2
1554 $LEFDE0:
1555 $LSFDE2:
1556 .4 byte $LEFDE2-$LASFDE2
1557 $LASFDE2:
1558 .4 byte $Lframe0
1559 .4 byte $LFB31
1560 .4 byte $LFE31-$LFB31
1561 .byte 0x4
1562 .4 byte $LCFI5-$LFB31
1563 .byte 0xe
1564 .uleb128 0x30
1565 .byte 0x4
1566 .4 byte $LCFI8-$LCFI5
1567 .byte 0x11
1568 .uleb128 0x1c
1569 .sleb128 -4
1570 .byte 0x11
1571 .uleb128 0x1e
1572 .sleb128 -3
1573 .byte 0x11
1574 .uleb128 0x40
1575 .sleb128 -2
1576 .byte 0x4
1577 .4 byte $LCFI9-$LCFI8
1578 .byte 0xc
1579 .uleb128 0x1e
1580 .uleb128 0x30
1581 .align 2
1582 $LEFDE2:

```

```

1583 $LSFDE4:
1584 .4 byte $LEFDE4-$LASFDE4
1585 $LASFDE4:
1586 .4 byte $Lframe0
1587 .4 byte $LFB33
1588 .4 byte $LFE33-$LFB33
1589 .byte 0x4
1590 .4 byte $LCFI10-$LFB33
1591 .byte 0xe
1592 .uleb128 0x30
1593 .byte 0x4
1594 .4 byte $LCFI13-$LCFI10
1595 .byte 0x11
1596 .uleb128 0x1c
1597 .sleb128 -4
1598 .byte 0x11
1599 .uleb128 0x1e
1600 .sleb128 -3
1601 .byte 0x11
1602 .uleb128 0x40
1603 .sleb128 -2
1604 .byte 0x4
1605 .4 byte $LCFI14-$LCFI13
1606 .byte 0xc
1607 .uleb128 0x1e
1608 .uleb128 0x30
1609 .align 2
1610 $LEFDE4:
1611 $LSFDE6:
1612 .4 byte $LEFDE6-$LASFDE6
1613 $LASFDE6:
1614 .4 byte $Lframe0
1615 .4 byte $LFB35
1616 .4 byte $LFE35-$LFB35
1617 .byte 0x4
1618 .4 byte $LCFI15-$LFB35
1619 .byte 0xe
1620 .uleb128 0x30
1621 .byte 0x4
1622 .4 byte $LCFI19-$LCFI15
1623 .byte 0x11
1624 .uleb128 0x35
1625 .sleb128 -1
1626 .byte 0x11
1627 .uleb128 0x34
1628 .sleb128 -2
1629 .byte 0x11
1630 .uleb128 0x1c
1631 .sleb128 -6
1632 .byte 0x11

```

```

1633 .uleb128 0x1e
1634 .sleb128 -5
1635 .byte 0x11
1636 .uleb128 0x40
1637 .sleb128 -4
1638 .byte 0x4
1639 .4 byte $LCFI20-$LCFI19
1640 .byte 0xc
1641 .uleb128 0x1e
1642 .uleb128 0x30
1643 .align 2
1644 $LEFDE6:
1645 $LSFDE8:
1646 .4 byte $LEFDE8-$LASFDE8
1647 $LASFDE8:
1648 .4 byte $Lframe0
1649 .4 byte $LFB37
1650 .4 byte $LFE37-$LFB37
1651 .byte 0x4
1652 .4 byte $LCFI21-$LFB37
1653 .byte 0xe
1654 .uleb128 0x38
1655 .byte 0x4
1656 .4 byte $LCFI26-$LCFI21
1657 .byte 0x11
1658 .uleb128 0x35
1659 .sleb128 -1
1660 .byte 0x11
1661 .uleb128 0x34
1662 .sleb128 -2
1663 .byte 0x11
1664 .uleb128 0x10
1665 .sleb128 -6
1666 .byte 0x11
1667 .uleb128 0x1c
1668 .sleb128 -5
1669 .byte 0x11
1670 .uleb128 0x1e
1671 .sleb128 -4
1672 .byte 0x11
1673 .uleb128 0x40
1674 .sleb128 -3
1675 .byte 0x4
1676 .4 byte $LCFI27-$LCFI26
1677 .byte 0xc
1678 .uleb128 0x1e
1679 .uleb128 0x38
1680 .align 2
1681 $LEFDE8:
1682 $LSFDE10:

```

```

1683 .4 byte $LEFDE10-$LASFDE10
1684 $LASFDE10:
1685 .4 byte $Lframe0
1686 .4 byte $LFB39
1687 .4 byte $LFE39-$LFB39
1688 .byte 0x4
1689 .4 byte $LCFI28-$LFB39
1690 .byte 0xe
1691 .uleb128 0x30
1692 .byte 0x4
1693 .4 byte $LCFI31-$LCFI28
1694 .byte 0x11
1695 .uleb128 0x1c
1696 .sleb128 -4
1697 .byte 0x11
1698 .uleb128 0x1e
1699 .sleb128 -3
1700 .byte 0x11
1701 .uleb128 0x40
1702 .sleb128 -2
1703 .byte 0x4
1704 .4 byte $LCFI32-$LCFI31
1705 .byte 0xc
1706 .uleb128 0x1e
1707 .uleb128 0x30
1708 .align 2
1709 $LEFDE10:
1710 $LSFDE12:
1711 .4 byte $LEFDE12-$LASFDE12
1712 $LASFDE12:
1713 .4 byte $Lframe0
1714 .4 byte $LFB41
1715 .4 byte $LFE41-$LFB41
1716 .byte 0x4
1717 .4 byte $LCFI33-$LFB41
1718 .byte 0xe
1719 .uleb128 0x30
1720 .byte 0x4
1721 .4 byte $LCFI37-$LCFI33
1722 .byte 0x11
1723 .uleb128 0x10
1724 .sleb128 -4
1725 .byte 0x11
1726 .uleb128 0x1c
1727 .sleb128 -3
1728 .byte 0x11
1729 .uleb128 0x1e
1730 .sleb128 -2
1731 .byte 0x11
1732 .uleb128 0x40

```

```

1733 .sleb128 -1
1734 .byte 0x4
1735 .4 byte $LCFI38-$LCFI37
1736 .byte 0xc
1737 .uleb128 0x1e
1738 .uleb128 0x30
1739 .align 2
1740 $LEFDE12:
1741 $LSFDE14:
1742 .4 byte $LEFDE14-$LASFDE14
1743 $LASFDE14:
1744 .4 byte $Lframe0
1745 .4 byte $LFB43
1746 .4 byte $LFE43-$LFB43
1747 .byte 0x4
1748 .4 byte $LCFI39-$LFB43
1749 .byte 0xe
1750 .uleb128 0x88
1751 .byte 0x4
1752 .4 byte $LCFI42-$LCFI39
1753 .byte 0x11
1754 .uleb128 0x1c
1755 .sleb128 -4
1756 .byte 0x11
1757 .uleb128 0x1e
1758 .sleb128 -3
1759 .byte 0x11
1760 .uleb128 0x40
1761 .sleb128 -2
1762 .byte 0x4
1763 .4 byte $LCFI43-$LCFI42
1764 .byte 0xc
1765 .uleb128 0x1e
1766 .uleb128 0x88
1767 .align 2
1768 $LEFDE14:
1769 $LSFDE16:
1770 .4 byte $LEFDE16-$LASFDE16
1771 $LASFDE16:
1772 .4 byte $Lframe0
1773 .4 byte $LFB45
1774 .4 byte $LFE45-$LFB45
1775 .byte 0x4
1776 .4 byte $LCFI44-$LFB45
1777 .byte 0xe
1778 .uleb128 0xb0
1779 .byte 0x4
1780 .4 byte $LCFI47-$LCFI44
1781 .byte 0x11
1782 .uleb128 0x1c

```



```

1783 .sleb128 -4
1784 .byte 0x11
1785 .uleb128 0x1e
1786 .sleb128 -3
1787 .byte 0x11
1788 .uleb128 0x40
1789 .sleb128 -2
1790 .byte 0x4
1791 .4byte $LCFI48-$LCFI47
1792 .byte 0xc
1793 .uleb128 0x1e
1794 .uleb128 0xb0
1795 .align 2
1796 $LEFDE16:
1797 .align 0
1798 .text
1799 $Letext0:
1800 .section .debug_info
1801 .4byte 0x115e
1802 .2byte 0x2
1803 .4byte $Ldebug_abbrev0
1804 .byte 0x4
1805 .uleb128 0x1
1806 .4byte $Ldebug_line0
1807 .4byte $Letext0
1808 .4byte $Ltext0
1809 .4byte $LC285
1810 .4byte $LC286
1811 .4byte $LC287
1812 .byte 0x1
1813 .uleb128 0x2
1814 .4byte $LC29
1815 .byte 0x2
1816 .byte 0x30
1817 .4byte 0x30
1818 .uleb128 0x3
1819 .4byte $LC31
1820 .byte 0x1
1821 .byte 0x6
1822 .uleb128 0x2
1823 .4byte $LC30
1824 .byte 0x2
1825 .byte 0x31
1826 .4byte 0x42
1827 .uleb128 0x3
1828 .4byte $LC32
1829 .byte 0x1
1830 .byte 0x8
1831 .uleb128 0x2
1832 .4byte $LC33

```

1833 .byte 0x2
1834 .byte 0x32
1835 .4byte 0x54
1836 .uleb128 0x3
1837 .4byte \$LC34
1838 .byte 0x2
1839 .byte 0x5
1840 .uleb128 0x2
1841 .4byte \$LC35
1842 .byte 0x2
1843 .byte 0x33
1844 .4byte 0x66
1845 .uleb128 0x3
1846 .4byte \$LC36
1847 .byte 0x2
1848 .byte 0x7
1849 .uleb128 0x2
1850 .4byte \$LC37
1851 .byte 0x2
1852 .byte 0x34
1853 .4byte 0x78
1854 .uleb128 0x4
1855 .ascii "int\000"
1856 .byte 0x4
1857 .byte 0x5
1858 .uleb128 0x2
1859 .4byte \$LC38
1860 .byte 0x2
1861 .byte 0x35
1862 .4byte 0x8a
1863 .uleb128 0x3
1864 .4byte \$LC39
1865 .byte 0x4
1866 .byte 0x7
1867 .uleb128 0x2
1868 .4byte \$LC40
1869 .byte 0x2
1870 .byte 0x3e
1871 .4byte 0x9c
1872 .uleb128 0x3
1873 .4byte \$LC41
1874 .byte 0x8
1875 .byte 0x5
1876 .uleb128 0x2
1877 .4byte \$LC42
1878 .byte 0x2
1879 .byte 0x40
1880 .4byte 0xae
1881 .uleb128 0x3
1882 .4byte \$LC43

1883 .byte 0x8
1884 .byte 0x7
1885 .uleb128 0x2
1886 .4byte \$LC44
1887 .byte 0x2
1888 .byte 0x4b
1889 .4byte 0x78
1890 .uleb128 0x2
1891 .4byte \$LC45
1892 .byte 0x2
1893 .byte 0x4c
1894 .4byte 0x8a
1895 .uleb128 0x2
1896 .4byte \$LC46
1897 .byte 0x3
1898 .byte 0x2c
1899 .4byte 0xd6
1900 .uleb128 0x5
1901 .byte 0x4
1902 .4byte 0xdc
1903 .uleb128 0x3
1904 .4byte \$LC47
1905 .byte 0x1
1906 .byte 0x6
1907 .uleb128 0x2
1908 .4byte \$LC48
1909 .byte 0x3
1910 .byte 0x2d
1911 .4byte 0x7f
1912 .uleb128 0x2
1913 .4byte \$LC49
1914 .byte 0x3
1915 .byte 0x2e
1916 .4byte 0x7f
1917 .uleb128 0x2
1918 .4byte \$LC50
1919 .byte 0x3
1920 .byte 0x2f
1921 .4byte 0x5b
1922 .uleb128 0x2
1923 .4byte \$LC51
1924 .byte 0x3
1925 .byte 0x30
1926 .4byte 0x7f
1927 .uleb128 0x2
1928 .4byte \$LC52
1929 .byte 0x3
1930 .byte 0x31
1931 .4byte 0x91
1932 .uleb128 0x2

1933 .4 byte \$LC53
1934 .byte 0x3
1935 .byte 0x32
1936 .4 byte 0x6d
1937 .uleb128 0x2
1938 .4 byte \$LC54
1939 .byte 0x3
1940 .byte 0x33
1941 .4 byte 0x37
1942 .uleb128 0x2
1943 .4 byte \$LC55
1944 .byte 0x3
1945 .byte 0x34
1946 .4 byte 0x8a
1947 .uleb128 0x2
1948 .4 byte \$LC56
1949 .byte 0x3
1950 .byte 0x35
1951 .4 byte 0x7f
1952 .uleb128 0x2
1953 .4 byte \$LC57
1954 .byte 0x3
1955 .byte 0x36
1956 .4 byte 0xa3
1957 .uleb128 0x2
1958 .4 byte \$LC58
1959 .byte 0x3
1960 .byte 0x37
1961 .4 byte 0xa3
1962 .uleb128 0x6
1963 .4 byte 0x17b
1964 .byte 0x80
1965 .byte 0x4
1966 .byte 0x65
1967 .uleb128 0x7
1968 .4 byte \$LC59
1969 .byte 0x4
1970 .byte 0x63
1971 .4 byte 0x17b
1972 .uleb128 0x7
1973 .4 byte \$LC60
1974 .byte 0x4
1975 .byte 0x64
1976 .4 byte 0x91
1977 .byte 0x0
1978 .uleb128 0x8
1979 .4 byte 0x18b
1980 .4 byte 0xdc
1981 .uleb128 0x9
1982 .4 byte 0x18b

```

1983 .byte 0x7f
1984 .byte 0x0
1985 .uleb128 0x3
1986 .4byte $LC39
1987 .byte 0x4
1988 .byte 0x7
1989 .uleb128 0x2
1990 .4byte $LC61
1991 .byte 0x4
1992 .byte 0x65
1993 .4byte 0x15c
1994 .uleb128 0x2
1995 .4byte $LC62
1996 .byte 0x5
1997 .byte 0x2e
1998 .4byte 0x8a
1999 .uleb128 0xa
2000 .4byte 0x1c3
2001 .4byte $LC65
2002 .byte 0x8
2003 .byte 0x5
2004 .byte 0x3c
2005 .uleb128 0xb
2006 .4byte $LC63
2007 .byte 0x5
2008 .byte 0x3d
2009 .4byte 0x10f
2010 .byte 0x2
2011 .byte 0x10
2012 .uleb128 0x0
2013 .byte 0x0
2014 .uleb128 0x2
2015 .4byte $LC64
2016 .byte 0x5
2017 .byte 0x3e
2018 .4byte 0x1a8
2019 .uleb128 0xa
2020 .4byte 0x1f7
2021 .4byte $LC66
2022 .byte 0x8
2023 .byte 0x5
2024 .byte 0x4a
2025 .uleb128 0xb
2026 .4byte $LC67
2027 .byte 0x5
2028 .byte 0x4b
2029 .4byte 0x1f7
2030 .byte 0x2
2031 .byte 0x10
2032 .uleb128 0x0

```

```

2033 .uleb128 0xb
2034 .4byte $LC68
2035 .byte 0x5
2036 .byte 0x4c
2037 .4byte 0x78
2038 .byte 0x2
2039 .byte 0x10
2040 .uleb128 0x4
2041 .byte 0x0
2042 .uleb128 0x5
2043 .byte 0x4
2044 .4byte 0x42
2045 .uleb128 0xa
2046 .4byte 0x31f
2047 .4byte $LC69
2048 .byte 0x58
2049 .byte 0x5
2050 .byte 0x69
2051 .uleb128 0xc
2052 .ascii "_p\000"
2053 .byte 0x5
2054 .byte 0x6a
2055 .4byte 0x1f7
2056 .byte 0x2
2057 .byte 0x10
2058 .uleb128 0x0
2059 .uleb128 0xc
2060 .ascii "_r\000"
2061 .byte 0x5
2062 .byte 0x6b
2063 .4byte 0x78
2064 .byte 0x2
2065 .byte 0x10
2066 .uleb128 0x4
2067 .uleb128 0xc
2068 .ascii "_w\000"
2069 .byte 0x5
2070 .byte 0x6c
2071 .4byte 0x78
2072 .byte 0x2
2073 .byte 0x10
2074 .uleb128 0x8
2075 .uleb128 0xb
2076 .4byte $LC70
2077 .byte 0x5
2078 .byte 0x6d
2079 .4byte 0x54
2080 .byte 0x2
2081 .byte 0x10
2082 .uleb128 0xc

```

2083 .uleb128 0xb
2084 .4byte \$LC71
2085 .byte 0x5
2086 .byte 0x6e
2087 .4byte 0x54
2088 .byte 0x2
2089 .byte 0x10
2090 .uleb128 0xe
2091 .uleb128 0xc
2092 .ascii "_bf\000"
2093 .byte 0x5
2094 .byte 0x6f
2095 .4byte 0x1ce
2096 .byte 0x2
2097 .byte 0x10
2098 .uleb128 0x10
2099 .uleb128 0xb
2100 .4byte \$LC72
2101 .byte 0x5
2102 .byte 0x70
2103 .4byte 0x78
2104 .byte 0x2
2105 .byte 0x10
2106 .uleb128 0x18
2107 .uleb128 0xb
2108 .4byte \$LC73
2109 .byte 0x5
2110 .byte 0x73
2111 .4byte 0x31f
2112 .byte 0x2
2113 .byte 0x10
2114 .uleb128 0x1c
2115 .uleb128 0xb
2116 .4byte \$LC74
2117 .byte 0x5
2118 .byte 0x74
2119 .4byte 0x331
2120 .byte 0x2
2121 .byte 0x10
2122 .uleb128 0x20
2123 .uleb128 0xb
2124 .4byte \$LC75
2125 .byte 0x5
2126 .byte 0x75
2127 .4byte 0x351
2128 .byte 0x2
2129 .byte 0x10
2130 .uleb128 0x24
2131 .uleb128 0xb
2132 .4byte \$LC76

```

2133 .byte 0x5
2134 .byte 0x76
2135 .4byte 0x371
2136 .byte 0x2
2137 .byte 0x10
2138 .uleb128 0x28
2139 .uleb128 0xb
2140 .4byte $LC77
2141 .byte 0x5
2142 .byte 0x77
2143 .4byte 0x39c
2144 .byte 0x2
2145 .byte 0x10
2146 .uleb128 0x2c
2147 .uleb128 0xb
2148 .4byte $LC78
2149 .byte 0x5
2150 .byte 0x7a
2151 .4byte 0x1ce
2152 .byte 0x2
2153 .byte 0x10
2154 .uleb128 0x30
2155 .uleb128 0xc
2156 .ascii "_up\000"
2157 .byte 0x5
2158 .byte 0x7d
2159 .4byte 0x1f7
2160 .byte 0x2
2161 .byte 0x10
2162 .uleb128 0x38
2163 .uleb128 0xc
2164 .ascii "_ur\000"
2165 .byte 0x5
2166 .byte 0x7e
2167 .4byte 0x78
2168 .byte 0x2
2169 .byte 0x10
2170 .uleb128 0x3c
2171 .uleb128 0xb
2172 .4byte $LC79
2173 .byte 0x5
2174 .byte 0x81
2175 .4byte 0x3a2
2176 .byte 0x2
2177 .byte 0x10
2178 .uleb128 0x40
2179 .uleb128 0xb
2180 .4byte $LC80
2181 .byte 0x5
2182 .byte 0x82

```



```

2183 .4byte 0x3b2
2184 .byte 0x2
2185 .byte 0x10
2186 .uleb128 0x43
2187 .uleb128 0xc
2188 .ascii "_lb\000"
2189 .byte 0x5
2190 .byte 0x85
2191 .4byte 0x1ce
2192 .byte 0x2
2193 .byte 0x10
2194 .uleb128 0x44
2195 .uleb128 0xb
2196 .4byte $LC81
2197 .byte 0x5
2198 .byte 0x88
2199 .4byte 0x78
2200 .byte 0x2
2201 .byte 0x10
2202 .uleb128 0x4c
2203 .uleb128 0xb
2204 .4byte $LC82
2205 .byte 0x5
2206 .byte 0x89
2207 .4byte 0x1c3
2208 .byte 0x2
2209 .byte 0x10
2210 .uleb128 0x50
2211 .byte 0x0
2212 .uleb128 0xd
2213 .byte 0x4
2214 .uleb128 0xe
2215 .4byte 0x331
2216 .byte 0x1
2217 .4byte 0x78
2218 .uleb128 0xf
2219 .4byte 0x31f
2220 .byte 0x0
2221 .uleb128 0x5
2222 .byte 0x4
2223 .4byte 0x321
2224 .uleb128 0xe
2225 .4byte 0x351
2226 .byte 0x1
2227 .4byte 0x78
2228 .uleb128 0xf
2229 .4byte 0x31f
2230 .uleb128 0xf
2231 .4byte 0xd6
2232 .uleb128 0xf

```

2233 .4 byte 0x78
2234 .byte 0x0
2235 .uleb128 0x5
2236 .byte 0x4
2237 .4 byte 0x337
2238 .uleb128 0xe
2239 .4 byte 0x371
2240 .byte 0x1
2241 .4 byte 0x1c3
2242 .uleb128 0xf
2243 .4 byte 0x31f
2244 .uleb128 0xf
2245 .4 byte 0x1c3
2246 .uleb128 0xf
2247 .4 byte 0x78
2248 .byte 0x0
2249 .uleb128 0x5
2250 .byte 0x4
2251 .4 byte 0x357
2252 .uleb128 0xe
2253 .4 byte 0x391
2254 .byte 0x1
2255 .4 byte 0x78
2256 .uleb128 0xf
2257 .4 byte 0x31f
2258 .uleb128 0xf
2259 .4 byte 0x391
2260 .uleb128 0xf
2261 .4 byte 0x78
2262 .byte 0x0
2263 .uleb128 0x5
2264 .byte 0x4
2265 .4 byte 0x397
2266 .uleb128 0x10
2267 .4 byte 0xdc
2268 .uleb128 0x5
2269 .byte 0x4
2270 .4 byte 0x377
2271 .uleb128 0x8
2272 .4 byte 0x3b2
2273 .4 byte 0x42
2274 .uleb128 0x9
2275 .4 byte 0x18b
2276 .byte 0x2
2277 .byte 0x0
2278 .uleb128 0x8
2279 .4 byte 0x3c2
2280 .4 byte 0x42
2281 .uleb128 0x9
2282 .4 byte 0x18b

2283 .byte 0x0
2284 .byte 0x0
2285 .uleb128 0x2
2286 .4byte \$LC83
2287 .byte 0x5
2288 .byte 0x8a
2289 .4byte 0x1fd
2290 .uleb128 0x11
2291 .4byte \$LC84
2292 .byte 0x5
2293 .2byte 0x160
2294 .4byte 0x10f
2295 .uleb128 0x2
2296 .4byte \$LC85
2297 .byte 0x6
2298 .byte 0x3b
2299 .4byte 0x3e4
2300 .uleb128 0x3
2301 .4byte \$LC86
2302 .byte 0x4
2303 .byte 0x5
2304 .uleb128 0x2
2305 .4byte \$LC87
2306 .byte 0x6
2307 .byte 0x3c
2308 .4byte 0x3f6
2309 .uleb128 0x3
2310 .4byte \$LC88
2311 .byte 0x4
2312 .byte 0x7
2313 .uleb128 0x2
2314 .4byte \$LC89
2315 .byte 0x6
2316 .byte 0x3d
2317 .4byte 0x3e4
2318 .uleb128 0x2
2319 .4byte \$LC90
2320 .byte 0x6
2321 .byte 0x46
2322 .4byte 0x3f6
2323 .uleb128 0x2
2324 .4byte \$LC91
2325 .byte 0x6
2326 .byte 0x47
2327 .4byte 0x3f6
2328 .uleb128 0x2
2329 .4byte \$LC92
2330 .byte 0x6
2331 .byte 0x49
2332 .4byte 0x3f6

2333 .uleb128 0x2
2334 .4 byte \$LC93
2335 .byte 0x6
2336 .byte 0x4a
2337 .4 byte 0x3f6
2338 .uleb128 0x2
2339 .4 byte \$LC94
2340 .byte 0x6
2341 .byte 0x51
2342 .4 byte 0x3e4
2343 .uleb128 0x2
2344 .4 byte \$LC95
2345 .byte 0x6
2346 .byte 0x5a
2347 .4 byte 0x44a
2348 .uleb128 0x12
2349 .4 byte 0x78
2350 .uleb128 0x2
2351 .4 byte \$LC96
2352 .byte 0x7
2353 .byte 0x36
2354 .4 byte 0x25
2355 .uleb128 0x2
2356 .4 byte \$LC97
2357 .byte 0x7
2358 .byte 0x3b
2359 .4 byte 0x37
2360 .uleb128 0x2
2361 .4 byte \$LC98
2362 .byte 0x7
2363 .byte 0x40
2364 .4 byte 0x49
2365 .uleb128 0x2
2366 .4 byte \$LC99
2367 .byte 0x7
2368 .byte 0x45
2369 .4 byte 0x5b
2370 .uleb128 0x2
2371 .4 byte \$LC100
2372 .byte 0x7
2373 .byte 0x4a
2374 .4 byte 0x6d
2375 .uleb128 0x2
2376 .4 byte \$LC101
2377 .byte 0x7
2378 .byte 0x4f
2379 .4 byte 0x7f
2380 .uleb128 0x2
2381 .4 byte \$LC102
2382 .byte 0x7

2383 .byte 0x54
2384 .4byte 0x91
2385 .uleb128 0x2
2386 .4byte \$LC103
2387 .byte 0x7
2388 .byte 0x59
2389 .4byte 0xa3
2390 .uleb128 0x2
2391 .4byte \$LC104
2392 .byte 0x7
2393 .byte 0x5d
2394 .4byte 0x37
2395 .uleb128 0x2
2396 .4byte \$LC105
2397 .byte 0x7
2398 .byte 0x5e
2399 .4byte 0x5b
2400 .uleb128 0x2
2401 .4byte \$LC106
2402 .byte 0x7
2403 .byte 0x5f
2404 .4byte 0x7f
2405 .uleb128 0x2
2406 .4byte \$LC107
2407 .byte 0x7
2408 .byte 0x60
2409 .4byte 0xa3
2410 .uleb128 0x2
2411 .4byte \$LC108
2412 .byte 0x8
2413 .byte 0x3a
2414 .4byte 0xee
2415 .uleb128 0x2
2416 .4byte \$LC109
2417 .byte 0x8
2418 .byte 0x3f
2419 .4byte 0xf9
2420 .uleb128 0x2
2421 .4byte \$LC110
2422 .byte 0x7
2423 .byte 0x65
2424 .4byte 0x42
2425 .uleb128 0x2
2426 .4byte \$LC111
2427 .byte 0x7
2428 .byte 0x66
2429 .4byte 0x66
2430 .uleb128 0x2
2431 .4byte \$LC112
2432 .byte 0x7

2433 .byte 0x67
2434 .4byte 0x8a
2435 .uleb128 0x2
2436 .4byte \$LC113
2437 .byte 0x7
2438 .byte 0x68
2439 .4byte 0x3f6
2440 .uleb128 0x2
2441 .4byte \$LC114
2442 .byte 0x7
2443 .byte 0x6a
2444 .4byte 0x42
2445 .uleb128 0x2
2446 .4byte \$LC115
2447 .byte 0x7
2448 .byte 0x6b
2449 .4byte 0x66
2450 .uleb128 0x2
2451 .4byte \$LC116
2452 .byte 0x7
2453 .byte 0x6c
2454 .4byte 0x8a
2455 .uleb128 0x2
2456 .4byte \$LC117
2457 .byte 0x7
2458 .byte 0x6d
2459 .4byte 0x3f6
2460 .uleb128 0x2
2461 .4byte \$LC118
2462 .byte 0x7
2463 .byte 0x6f
2464 .4byte 0x50a
2465 .uleb128 0x2
2466 .4byte \$LC119
2467 .byte 0x7
2468 .byte 0x72
2469 .4byte 0xa3
2470 .uleb128 0x2
2471 .4byte \$LC120
2472 .byte 0x7
2473 .byte 0x73
2474 .4byte 0x91
2475 .uleb128 0x2
2476 .4byte \$LC121
2477 .byte 0x7
2478 .byte 0x74
2479 .4byte 0x56d
2480 .uleb128 0x5
2481 .byte 0x4
2482 .4byte 0x557

```

2483 .uleb128 0x2
2484 .4byte $LC122
2485 .byte 0x7
2486 .byte 0x80
2487 .4byte 0x91
2488 .uleb128 0x2
2489 .4byte $LC123
2490 .byte 0x7
2491 .byte 0x81
2492 .4byte 0xa3
2493 .uleb128 0x2
2494 .4byte $LC124
2495 .byte 0x7
2496 .byte 0x83
2497 .4byte 0x91
2498 .uleb128 0x2
2499 .4byte $LC125
2500 .byte 0x7
2501 .byte 0x84
2502 .4byte 0x7f
2503 .uleb128 0x2
2504 .4byte $LC126
2505 .byte 0x7
2506 .byte 0x87
2507 .4byte 0x146
2508 .uleb128 0x2
2509 .4byte $LC127
2510 .byte 0x7
2511 .byte 0x8c
2512 .4byte 0x151
2513 .uleb128 0x2
2514 .4byte $LC128
2515 .byte 0x7
2516 .byte 0x91
2517 .4byte 0xcb
2518 .uleb128 0x2
2519 .4byte $LC129
2520 .byte 0x7
2521 .byte 0x99
2522 .4byte 0x91
2523 .uleb128 0x2
2524 .4byte $LC130
2525 .byte 0x7
2526 .byte 0x9c
2527 .4byte 0x7f
2528 .uleb128 0x2
2529 .4byte $LC131
2530 .byte 0x7
2531 .byte 0x9d
2532 .4byte 0x7f

```

```

2533 .uleb128 0x2
2534 .4byte $LC132
2535 .byte 0x7
2536 .byte 0xa0
2537 .4byte 0xe3
2538 .uleb128 0x2
2539 .4byte $LC133
2540 .byte 0x7
2541 .byte 0xa4
2542 .4byte 0x7f
2543 .uleb128 0x2
2544 .4byte $LC134
2545 .byte 0x7
2546 .byte 0xa5
2547 .4byte 0x7f
2548 .uleb128 0x2
2549 .4byte $LC135
2550 .byte 0x7
2551 .byte 0xa6
2552 .4byte 0x3e4
2553 .uleb128 0x2
2554 .4byte $LC136
2555 .byte 0x7
2556 .byte 0xa9
2557 .4byte 0x104
2558 .uleb128 0x2
2559 .4byte $LC137
2560 .byte 0x7
2561 .byte 0xad
2562 .4byte 0x7f
2563 .uleb128 0x2
2564 .4byte $LC138
2565 .byte 0x7
2566 .byte 0xb5
2567 .4byte 0x11a
2568 .uleb128 0x2
2569 .4byte $LC139
2570 .byte 0x7
2571 .byte 0xb8
2572 .4byte 0x6d
2573 .uleb128 0x2
2574 .4byte $LC140
2575 .byte 0x7
2576 .byte 0xb9
2577 .4byte 0x557
2578 .uleb128 0x2
2579 .4byte $LC141
2580 .byte 0x7
2581 .byte 0xba
2582 .4byte 0x6d

```



```

2583 .uleb128 0x2
2584 .4byte $LC142
2585 .byte 0x7
2586 .byte 0xbb
2587 .4byte 0x6d
2588 .uleb128 0x2
2589 .4byte $LC143
2590 .byte 0x7
2591 .byte 0xbe
2592 .4byte 0x13b
2593 .uleb128 0x2
2594 .4byte $LC144
2595 .byte 0x7
2596 .byte 0xc2
2597 .4byte 0x6d
2598 .uleb128 0x2
2599 .4byte $LC145
2600 .byte 0x7
2601 .byte 0xfc
2602 .4byte 0x3f6
2603 .uleb128 0x11
2604 .4byte $LC146
2605 .byte 0x7
2606 .2byte 0x107
2607 .4byte 0x78
2608 .uleb128 0x11
2609 .4byte $LC147
2610 .byte 0x7
2611 .2byte 0x10c
2612 .4byte 0x3e4
2613 .uleb128 0x11
2614 .4byte $LC148
2615 .byte 0x7
2616 .2byte 0x111
2617 .4byte 0x78
2618 .uleb128 0x11
2619 .4byte $LC149
2620 .byte 0x7
2621 .2byte 0x116
2622 .4byte 0x78
2623 .uleb128 0x11
2624 .4byte $LC150
2625 .byte 0x7
2626 .2byte 0x11b
2627 .4byte 0x78
2628 .uleb128 0x11
2629 .4byte $LC151
2630 .byte 0x7
2631 .2byte 0x120
2632 .4byte 0x8a

```

```

2633 .uleb128 0x11
2634 .4 byte $LC152
2635 .byte 0x7
2636 .2 byte 0x12d
2637 .4 byte 0x6d
2638 .uleb128 0x13
2639 .4 byte 0x6ec
2640 .4 byte $LC153
2641 .byte 0x20
2642 .byte 0x7
2643 .2 byte 0x142
2644 .uleb128 0x14
2645 .4 byte $LC154
2646 .byte 0x7
2647 .2 byte 0x143
2648 .4 byte 0x6ec
2649 .byte 0x2
2650 .byte 0x10
2651 .uleb128 0x0
2652 .byte 0x0
2653 .uleb128 0x8
2654 .4 byte 0x6fc
2655 .4 byte 0x6c3
2656 .uleb128 0x9
2657 .4 byte 0x18b
2658 .byte 0x7
2659 .byte 0x0
2660 .uleb128 0x11
2661 .4 byte $LC153
2662 .byte 0x7
2663 .2 byte 0x144
2664 .4 byte 0x6cf
2665 .uleb128 0x2
2666 .4 byte $LC155
2667 .byte 0x9
2668 .byte 0x2e
2669 .4 byte 0x44a
2670 .uleb128 0xa
2671 .4 byte 0x73c
2672 .4 byte $LC156
2673 .byte 0x8
2674 .byte 0x9
2675 .byte 0x39
2676 .uleb128 0xb
2677 .4 byte $LC157
2678 .byte 0x9
2679 .byte 0x39
2680 .4 byte 0x742
2681 .byte 0x2
2682 .byte 0x10

```

```

2683 .uleb128 0x0
2684 .uleb128 0xb
2685 .4byte $LC158
2686 .byte 0x9
2687 .byte 0x39
2688 .4byte 0x748
2689 .byte 0x2
2690 .byte 0x10
2691 .uleb128 0x4
2692 .byte 0x0
2693 .uleb128 0x15
2694 .4byte $LC283
2695 .byte 0x1
2696 .uleb128 0x5
2697 .byte 0x4
2698 .4byte 0x73c
2699 .uleb128 0x5
2700 .byte 0x4
2701 .4byte 0x742
2702 .uleb128 0x2
2703 .4byte $LC159
2704 .byte 0x9
2705 .byte 0x47
2706 .4byte 0x742
2707 .uleb128 0x2
2708 .4byte $LC160
2709 .byte 0x9
2710 .byte 0x48
2711 .4byte 0x764
2712 .uleb128 0xa
2713 .4byte 0x79b
2714 .4byte $LC161
2715 .byte 0xc
2716 .byte 0x9
2717 .byte 0x3c
2718 .uleb128 0xb
2719 .4byte $LC162
2720 .byte 0x9
2721 .byte 0x56
2722 .4byte 0x8a
2723 .byte 0x2
2724 .byte 0x10
2725 .uleb128 0x0
2726 .uleb128 0xb
2727 .4byte $LC163
2728 .byte 0x9
2729 .byte 0x58
2730 .4byte 0x78
2731 .byte 0x2
2732 .byte 0x10

```

2733 .uleb128 0x4
2734 .uleb128 0xb
2735 .4byte \$LC164
2736 .byte 0x9
2737 .byte 0x59
2738 .4byte 0x31f
2739 .byte 0x2
2740 .byte 0x10
2741 .uleb128 0x8
2742 .byte 0x0
2743 .uleb128 0x2
2744 .4byte \$LC165
2745 .byte 0x9
2746 .byte 0x49
2747 .4byte 0x7a6
2748 .uleb128 0xa
2749 .4byte 0x807
2750 .4byte \$LC166
2751 .byte 0x1c
2752 .byte 0x9
2753 .byte 0x3d
2754 .uleb128 0xb
2755 .4byte \$LC167
2756 .byte 0x9
2757 .byte 0x5d
2758 .4byte 0x8a
2759 .byte 0x2
2760 .byte 0x10
2761 .uleb128 0x0
2762 .uleb128 0xb
2763 .4byte \$LC168
2764 .byte 0x9
2765 .byte 0x66
2766 .4byte 0x44a
2767 .byte 0x2
2768 .byte 0x10
2769 .uleb128 0x4
2770 .uleb128 0xb
2771 .4byte \$LC169
2772 .byte 0x9
2773 .byte 0x69
2774 .4byte 0x44a
2775 .byte 0x2
2776 .byte 0x10
2777 .uleb128 0x8
2778 .uleb128 0xb
2779 .4byte \$LC170
2780 .byte 0x9
2781 .byte 0x6a
2782 .4byte 0x74e

2783 .byte 0x2
2784 .byte 0x10
2785 .uleb128 0xc
2786 .uleb128 0xb
2787 .4byte \$LC171
2788 .byte 0x9
2789 .byte 0x6b
2790 .4byte 0x713
2791 .byte 0x2
2792 .byte 0x10
2793 .uleb128 0x10
2794 .uleb128 0xb
2795 .4byte \$LC172
2796 .byte 0x9
2797 .byte 0x6c
2798 .4byte 0x31f
2799 .byte 0x2
2800 .byte 0x10
2801 .uleb128 0x18
2802 .byte 0x0
2803 .uleb128 0x2
2804 .4byte \$LC173
2805 .byte 0x9
2806 .byte 0x4a
2807 .4byte 0x812
2808 .uleb128 0xa
2809 .4byte 0x83b
2810 .4byte \$LC174
2811 .byte 0x8
2812 .byte 0x9
2813 .byte 0x3e
2814 .uleb128 0xb
2815 .4byte \$LC175
2816 .byte 0x9
2817 .byte 0x7c
2818 .4byte 0x8a
2819 .byte 0x2
2820 .byte 0x10
2821 .uleb128 0x0
2822 .uleb128 0xb
2823 .4byte \$LC176
2824 .byte 0x9
2825 .byte 0x7d
2826 .4byte 0x31f
2827 .byte 0x2
2828 .byte 0x10
2829 .uleb128 0x4
2830 .byte 0x0
2831 .uleb128 0x2
2832 .4byte \$LC177

2833 .byte 0x9
2834 .byte 0x4b
2835 .4byte 0x846
2836 .uleb128 0xa
2837 .4byte 0x899
2838 .4byte \$LC178
2839 .byte 0x18
2840 .byte 0x9
2841 .byte 0x3f
2842 .uleb128 0xb
2843 .4byte \$LC179
2844 .byte 0x9
2845 .byte 0x85
2846 .4byte 0x8a
2847 .byte 0x2
2848 .byte 0x10
2849 .uleb128 0x0
2850 .uleb128 0xb
2851 .4byte \$LC180
2852 .byte 0x9
2853 .byte 0x88
2854 .4byte 0x44a
2855 .byte 0x2
2856 .byte 0x10
2857 .uleb128 0x4
2858 .uleb128 0xb
2859 .4byte \$LC181
2860 .byte 0x9
2861 .byte 0x89
2862 .4byte 0x713
2863 .byte 0x2
2864 .byte 0x10
2865 .uleb128 0x8
2866 .uleb128 0xb
2867 .4byte \$LC182
2868 .byte 0x9
2869 .byte 0x8b
2870 .4byte 0xaaa
2871 .byte 0x2
2872 .byte 0x10
2873 .uleb128 0x10
2874 .uleb128 0xb
2875 .4byte \$LC183
2876 .byte 0x9
2877 .byte 0x8c
2878 .4byte 0x31f
2879 .byte 0x2
2880 .byte 0x10
2881 .uleb128 0x14
2882 .byte 0x0

```

2883 .uleb128 0x2
2884 .4byte $LC184
2885 .byte 0x9
2886 .byte 0x4c
2887 .4byte 0x8a4
2888 .uleb128 0xa
2889 .4byte 0x8cd
2890 .4byte $LC185
2891 .byte 0x8
2892 .byte 0x9
2893 .byte 0x40
2894 .uleb128 0xb
2895 .4byte $LC186
2896 .byte 0x9
2897 .byte 0x9a
2898 .4byte 0x8a
2899 .byte 0x2
2900 .byte 0x10
2901 .uleb128 0x0
2902 .uleb128 0xb
2903 .4byte $LC187
2904 .byte 0x9
2905 .byte 0x9b
2906 .4byte 0x31f
2907 .byte 0x2
2908 .byte 0x10
2909 .uleb128 0x4
2910 .byte 0x0
2911 .uleb128 0x2
2912 .4byte $LC188
2913 .byte 0x9
2914 .byte 0x4d
2915 .4byte 0x8d8
2916 .uleb128 0xa
2917 .4byte 0x901
2918 .4byte $LC189
2919 .byte 0x20
2920 .byte 0x9
2921 .byte 0x4d
2922 .uleb128 0xb
2923 .4byte $LC190
2924 .byte 0x9
2925 .byte 0xa2
2926 .4byte 0x79b
2927 .byte 0x2
2928 .byte 0x10
2929 .uleb128 0x0
2930 .uleb128 0xb
2931 .4byte $LC191
2932 .byte 0x9

```

2933 .byte 0xa3
2934 .4 byte 0x78
2935 .byte 0x2
2936 .byte 0x10
2937 .uleb128 0x1c
2938 .byte 0x0
2939 .uleb128 0x2
2940 .4 byte \$LC192
2941 .byte 0x9
2942 .byte 0x4e
2943 .4 byte 0x90c
2944 .uleb128 0xa
2945 .4 byte 0x943
2946 .4 byte \$LC193
2947 .byte 0xc
2948 .byte 0x9
2949 .byte 0x4e
2950 .uleb128 0xb
2951 .4 byte \$LC194
2952 .byte 0x9
2953 .byte 0xa9
2954 .4 byte 0x8a
2955 .byte 0x2
2956 .byte 0x10
2957 .uleb128 0x0
2958 .uleb128 0xb
2959 .4 byte \$LC195
2960 .byte 0x9
2961 .byte 0xaa
2962 .4 byte 0x44a
2963 .byte 0x2
2964 .byte 0x10
2965 .uleb128 0x4
2966 .uleb128 0xb
2967 .4 byte \$LC196
2968 .byte 0x9
2969 .byte 0xab
2970 .4 byte 0x78
2971 .byte 0x2
2972 .byte 0x10
2973 .uleb128 0x8
2974 .byte 0x0
2975 .uleb128 0x2
2976 .4 byte \$LC197
2977 .byte 0x9
2978 .byte 0x4f
2979 .4 byte 0x94e
2980 .uleb128 0xa
2981 .4 byte 0x9bd
2982 .4 byte \$LC198

2983 .byte 0x24
2984 .byte 0x9
2985 .byte 0x42
2986 .uleb128 0xb
2987 .4byte \$LC199
2988 .byte 0x9
2989 .byte 0xb9
2990 .4byte 0x8a
2991 .byte 0x2
2992 .byte 0x10
2993 .uleb128 0x0
2994 .uleb128 0xb
2995 .4byte \$LC200
2996 .byte 0x9
2997 .byte 0xbc
2998 .4byte 0x44a
2999 .byte 0x2
3000 .byte 0x10
3001 .uleb128 0x4
3002 .uleb128 0xb
3003 .4byte \$LC201
3004 .byte 0x9
3005 .byte 0xbe
3006 .4byte 0x713
3007 .byte 0x2
3008 .byte 0x10
3009 .uleb128 0x8
3010 .uleb128 0xb
3011 .4byte \$LC202
3012 .byte 0x9
3013 .byte 0xbf
3014 .4byte 0x713
3015 .byte 0x2
3016 .byte 0x10
3017 .uleb128 0x10
3018 .uleb128 0xb
3019 .4byte \$LC203
3020 .byte 0x9
3021 .byte 0xc0
3022 .4byte 0x8a
3023 .byte 0x2
3024 .byte 0x10
3025 .uleb128 0x18
3026 .uleb128 0xb
3027 .4byte \$LC204
3028 .byte 0x9
3029 .byte 0xc1
3030 .4byte 0x74e
3031 .byte 0x2
3032 .byte 0x10

```

3033 .uleb128 0x1c
3034 .uleb128 0xb
3035 .4byte $LC205
3036 .byte 0x9
3037 .byte 0xc2
3038 .4byte 0x31f
3039 .byte 0x2
3040 .byte 0x10
3041 .uleb128 0x20
3042 .byte 0x0
3043 .uleb128 0x2
3044 .4byte $LC206
3045 .byte 0x9
3046 .byte 0x50
3047 .4byte 0x9c8
3048 .uleb128 0xa
3049 .4byte 0x9f1
3050 .4byte $LC207
3051 .byte 0x8
3052 .byte 0x9
3053 .byte 0x43
3054 .uleb128 0xb
3055 .4byte $LC208
3056 .byte 0x9
3057 .byte 0xd2
3058 .4byte 0x8a
3059 .byte 0x2
3060 .byte 0x10
3061 .uleb128 0x0
3062 .uleb128 0xb
3063 .4byte $LC209
3064 .byte 0x9
3065 .byte 0xd3
3066 .4byte 0x31f
3067 .byte 0x2
3068 .byte 0x10
3069 .uleb128 0x4
3070 .byte 0x0
3071 .uleb128 0x2
3072 .4byte $LC210
3073 .byte 0x9
3074 .byte 0x51
3075 .4byte 0x9fc
3076 .uleb128 0xa
3077 .4byte 0xa6b
3078 .4byte $LC211
3079 .byte 0x20
3080 .byte 0x9
3081 .byte 0x44
3082 .uleb128 0xb

```

3083 .4 byte \$LC212
3084 .byte 0x9
3085 .byte 0xda
3086 .4 byte 0x8a
3087 .byte 0x2
3088 .byte 0x10
3089 .uleb128 0x0
3090 .uleb128 0xb
3091 .4 byte \$LC213
3092 .byte 0x9
3093 .byte 0xdd
3094 .4 byte 0x44a
3095 .byte 0x2
3096 .byte 0x10
3097 .uleb128 0x4
3098 .uleb128 0xb
3099 .4 byte \$LC214
3100 .byte 0x9
3101 .byte 0xdf
3102 .4 byte 0x713
3103 .byte 0x2
3104 .byte 0x10
3105 .uleb128 0x8
3106 .uleb128 0xb
3107 .4 byte \$LC215
3108 .byte 0x9
3109 .byte 0xe0
3110 .4 byte 0x8a
3111 .byte 0x2
3112 .byte 0x10
3113 .uleb128 0x10
3114 .uleb128 0xb
3115 .4 byte \$LC216
3116 .byte 0x9
3117 .byte 0xe1
3118 .4 byte 0x8a
3119 .byte 0x2
3120 .byte 0x10
3121 .uleb128 0x14
3122 .uleb128 0xb
3123 .4 byte \$LC217
3124 .byte 0x9
3125 .byte 0xe2
3126 .4 byte 0x8a
3127 .byte 0x2
3128 .byte 0x10
3129 .uleb128 0x18
3130 .uleb128 0xb
3131 .4 byte \$LC218
3132 .byte 0x9

```

3133 .byte 0xe4
3134 .4byte 0x31f
3135 .byte 0x2
3136 .byte 0x10
3137 .uleb128 0x1c
3138 .byte 0x0
3139 .uleb128 0x2
3140 .4byte $LC219
3141 .byte 0x9
3142 .byte 0x52
3143 .4byte 0xa76
3144 .uleb128 0xa
3145 .4byte 0xa9f
3146 .4byte $LC220
3147 .byte 0x8
3148 .byte 0x9
3149 .byte 0x45
3150 .uleb128 0xb
3151 .4byte $LC221
3152 .byte 0x9
3153 .byte 0xeb
3154 .4byte 0x8a
3155 .byte 0x2
3156 .byte 0x10
3157 .uleb128 0x0
3158 .uleb128 0xb
3159 .4byte $LC222
3160 .byte 0x9
3161 .byte 0xec
3162 .4byte 0x31f
3163 .byte 0x2
3164 .byte 0x10
3165 .uleb128 0x4
3166 .byte 0x0
3167 .uleb128 0x2
3168 .4byte $LC223
3169 .byte 0x9
3170 .byte 0x53
3171 .4byte 0x78
3172 .uleb128 0x5
3173 .byte 0x4
3174 .4byte 0x79b
3175 .uleb128 0x2
3176 .4byte $LC224
3177 .byte 0xa
3178 .byte 0x34
3179 .4byte 0x78
3180 .uleb128 0x16
3181 .4byte 0xae0
3182 .byte 0x8

```

```

3183 .byte 0xa
3184 .byte 0x3b
3185 .uleb128 0xb
3186 .4byte $LC225
3187 .byte 0xa
3188 .byte 0x39
3189 .4byte 0x78
3190 .byte 0x2
3191 .byte 0x10
3192 .uleb128 0x0
3193 .uleb128 0xc
3194 .ascii "rem\000"
3195 .byte 0xa
3196 .byte 0x3a
3197 .4byte 0x78
3198 .byte 0x2
3199 .byte 0x10
3200 .uleb128 0x4
3201 .byte 0x0
3202 .uleb128 0x2
3203 .4byte $LC226
3204 .byte 0xa
3205 .byte 0x3b
3206 .4byte 0xab
3207 .uleb128 0x16
3208 .4byte 0xb10
3209 .byte 0x8
3210 .byte 0xa
3211 .byte 0x40
3212 .uleb128 0xb
3213 .4byte $LC225
3214 .byte 0xa
3215 .byte 0x3e
3216 .4byte 0x3e4
3217 .byte 0x2
3218 .byte 0x10
3219 .uleb128 0x0
3220 .uleb128 0xc
3221 .ascii "rem\000"
3222 .byte 0xa
3223 .byte 0x3f
3224 .4byte 0x3e4
3225 .byte 0x2
3226 .byte 0x10
3227 .uleb128 0x4
3228 .byte 0x0
3229 .uleb128 0x2
3230 .4byte $LC227
3231 .byte 0xa
3232 .byte 0x40

```

```

3233 .4 byte 0xae
3234 .uleb128 0x16
3235 .4 byte 0xb4
3236 .byte 0x10
3237 .byte 0xa
3238 .byte 0x4a
3239 .uleb128 0xb
3240 .4 byte $LC225
3241 .byte 0xa
3242 .byte 0x47
3243 .4 byte 0x9c
3244 .byte 0x2
3245 .byte 0x10
3246 .uleb128 0x0
3247 .uleb128 0xc
3248 .ascii "rem\000"
3249 .byte 0xa
3250 .byte 0x49
3251 .4 byte 0x9c
3252 .byte 0x2
3253 .byte 0x10
3254 .uleb128 0x8
3255 .byte 0x0
3256 .uleb128 0x2
3257 .4 byte $LC228
3258 .byte 0xa
3259 .byte 0x4a
3260 .4 byte 0xb1b
3261 .uleb128 0x16
3262 .4 byte 0xb7
3263 .byte 0x10
3264 .byte 0xa
3265 .byte 0x51
3266 .uleb128 0xb
3267 .4 byte $LC225
3268 .byte 0xa
3269 .byte 0x4f
3270 .4 byte 0x557
3271 .byte 0x2
3272 .byte 0x10
3273 .uleb128 0x0
3274 .uleb128 0xc
3275 .ascii "rem\000"
3276 .byte 0xa
3277 .byte 0x50
3278 .4 byte 0x557
3279 .byte 0x2
3280 .byte 0x10
3281 .uleb128 0x8
3282 .byte 0x0

```

```

3283 .uleb128 0x2
3284 .4 byte $LC229
3285 .byte 0xa
3286 .byte 0x51
3287 .4 byte 0xb4b
3288 .uleb128 0x17
3289 .4 byte 0xb9e
3290 .4 byte $LC233
3291 .byte 0x4
3292 .byte 0xb
3293 .byte 0x18
3294 .uleb128 0x7
3295 .4 byte $LC230
3296 .byte 0xb
3297 .byte 0x19
3298 .4 byte 0xb9e
3299 .uleb128 0x7
3300 .4 byte $LC231
3301 .byte 0xb
3302 .byte 0x1a
3303 .4 byte 0xbae
3304 .byte 0x0
3305 .uleb128 0x8
3306 .4 byte 0xbae
3307 .4 byte 0x42
3308 .uleb128 0x9
3309 .4 byte 0x18b
3310 .byte 0x3
3311 .byte 0x0
3312 .uleb128 0x3
3313 .4 byte $LC232
3314 .byte 0x4
3315 .byte 0x4
3316 .uleb128 0x17
3317 .4 byte 0xbd8
3318 .4 byte $LC234
3319 .byte 0x8
3320 .byte 0xb
3321 .byte 0x1d
3322 .uleb128 0x7
3323 .4 byte $LC230
3324 .byte 0xb
3325 .byte 0x1e
3326 .4 byte 0xbd8
3327 .uleb128 0x7
3328 .4 byte $LC231
3329 .byte 0xb
3330 .byte 0x1f
3331 .4 byte 0xbe8
3332 .byte 0x0

```

```

3333 .uleb128 0x8
3334 .4 byte 0xbe8
3335 .4 byte 0x42
3336 .uleb128 0x9
3337 .4 byte 0x18b
3338 .byte 0x7
3339 .byte 0x0
3340 .uleb128 0x3
3341 .4 byte $LC235
3342 .byte 0x8
3343 .byte 0x4
3344 .uleb128 0x17
3345 .4 byte 0xc12
3346 .4 byte $LC236
3347 .byte 0x8
3348 .byte 0xb
3349 .byte 0x22
3350 .uleb128 0x7
3351 .4 byte $LC230
3352 .byte 0xb
3353 .byte 0x23
3354 .4 byte 0xbd8
3355 .uleb128 0x7
3356 .4 byte $LC231
3357 .byte 0xb
3358 .byte 0x24
3359 .4 byte 0xc12
3360 .byte 0x0
3361 .uleb128 0x3
3362 .4 byte $LC237
3363 .byte 0x8
3364 .byte 0x4
3365 .uleb128 0x18
3366 .4 byte 0xc3e
3367 .4 byte $LC288
3368 .byte 0x4
3369 .byte 0xb
3370 .byte 0x81
3371 .uleb128 0x19
3372 .4 byte $LC238
3373 .sleb128 -1
3374 .uleb128 0x1a
3375 .4 byte $LC239
3376 .byte 0x0
3377 .uleb128 0x1a
3378 .4 byte $LC240
3379 .byte 0x1
3380 .uleb128 0x1a
3381 .4 byte $LC241
3382 .byte 0x2

```


3383 .byte 0x0
3384 .uleb128 0xa
3385 .4byte 0xc91
3386 .4byte \$LC242
3387 .byte 0x20
3388 .byte 0xb
3389 .byte 0x95
3390 .uleb128 0xb
3391 .4byte \$LC243
3392 .byte 0xb
3393 .byte 0x96
3394 .4byte 0x78
3395 .byte 0x2
3396 .byte 0x10
3397 .uleb128 0x0
3398 .uleb128 0xb
3399 .4byte \$LC244
3400 .byte 0xb
3401 .byte 0x97
3402 .4byte 0xd6
3403 .byte 0x2
3404 .byte 0x10
3405 .uleb128 0x4
3406 .uleb128 0xb
3407 .4byte \$LC245
3408 .byte 0xb
3409 .byte 0x98
3410 .4byte 0xbe8
3411 .byte 0x2
3412 .byte 0x10
3413 .uleb128 0x8
3414 .uleb128 0xb
3415 .4byte \$LC246
3416 .byte 0xb
3417 .byte 0x99
3418 .4byte 0xbe8
3419 .byte 0x2
3420 .byte 0x10
3421 .uleb128 0x10
3422 .uleb128 0xb
3423 .4byte \$LC247
3424 .byte 0xb
3425 .byte 0x9a
3426 .4byte 0xbe8
3427 .byte 0x2
3428 .byte 0x10
3429 .uleb128 0x18
3430 .byte 0x0
3431 .uleb128 0x16
3432 .4byte 0xcb6

```

3433 .byte 0x10
3434 .byte 0x1
3435 .byte 0xc
3436 .uleb128 0xb
3437 .4byte $LC248
3438 .byte 0x1
3439 .byte 0xa
3440 .4byte 0xbe8
3441 .byte 0x2
3442 .byte 0x10
3443 .uleb128 0x0
3444 .uleb128 0xb
3445 .4byte $LC249
3446 .byte 0x1
3447 .byte 0xb
3448 .4byte 0xbae
3449 .byte 0x2
3450 .byte 0x10
3451 .uleb128 0x8
3452 .byte 0x0
3453 .uleb128 0x2
3454 .4byte $LC250
3455 .byte 0x1
3456 .byte 0xc
3457 .4byte 0xc91
3458 .uleb128 0x1b
3459 .4byte 0xd34
3460 .byte 0x1
3461 .4byte $LC257
3462 .byte 0x1
3463 .byte 0xe
3464 .byte 0x1
3465 .4byte $LFB29
3466 .4byte $LFE29
3467 .4byte $LSFDE0
3468 .byte 0x1
3469 .byte 0x6e
3470 .uleb128 0x1c
3471 .4byte $LC251
3472 .byte 0x1
3473 .byte 0xe
3474 .4byte 0xd34
3475 .byte 0x3
3476 .byte 0x91
3477 .sleb128 96
3478 .uleb128 0x1c
3479 .4byte $LC252
3480 .byte 0x1
3481 .byte 0xe
3482 .4byte 0x78

```

```

3483 .byte 0x3
3484 .byte 0x91
3485 .sleb128 100
3486 .uleb128 0x1c
3487 .4byte $LC253
3488 .byte 0x1
3489 .byte 0xe
3490 .4byte 0x78
3491 .byte 0x3
3492 .byte 0x91
3493 .sleb128 104
3494 .uleb128 0x1d
3495 .4byte $LC254
3496 .byte 0x1
3497 .byte 0x11
3498 .4byte 0xd3a
3499 .byte 0x2
3500 .byte 0x91
3501 .sleb128 24
3502 .uleb128 0x1d
3503 .4byte $LC255
3504 .byte 0x1
3505 .byte 0x12
3506 .4byte 0xd3a
3507 .byte 0x2
3508 .byte 0x91
3509 .sleb128 40
3510 .uleb128 0x1d
3511 .4byte $LC256
3512 .byte 0x1
3513 .byte 0x13
3514 .4byte 0xd4a
3515 .byte 0x2
3516 .byte 0x91
3517 .sleb128 56
3518 .byte 0x0
3519 .uleb128 0x5
3520 .byte 0x4
3521 .4byte 0x3c2
3522 .uleb128 0x8
3523 .4byte 0xd4a
3524 .4byte 0xdc
3525 .uleb128 0x9
3526 .4byte 0x18b
3527 .byte 0x9
3528 .byte 0x0
3529 .uleb128 0x8
3530 .4byte 0xd5a
3531 .4byte 0xdc
3532 .uleb128 0x9

```

```

3533 .4 byte 0x18b
3534 .byte 0x13
3535 .byte 0x0
3536 .uleb128 0x1b
3537 .4 byte 0xddc
3538 .byte 0x1
3539 .4 byte $LC258
3540 .byte 0x1
3541 .byte 0x1d
3542 .byte 0x1
3543 .4 byte $LFB31
3544 .4 byte $LFE31
3545 .4 byte $LSFDE2
3546 .byte 0x1
3547 .byte 0x6e
3548 .uleb128 0x1c
3549 .4 byte $LC251
3550 .byte 0x1
3551 .byte 0x1d
3552 .4 byte 0xd34
3553 .byte 0x2
3554 .byte 0x91
3555 .sleb128 48
3556 .uleb128 0x1c
3557 .4 byte $LC259
3558 .byte 0x1
3559 .byte 0x1d
3560 .4 byte 0xddc
3561 .byte 0x2
3562 .byte 0x91
3563 .sleb128 52
3564 .uleb128 0x1c
3565 .4 byte $LC252
3566 .byte 0x1
3567 .byte 0x1d
3568 .4 byte 0x78
3569 .byte 0x2
3570 .byte 0x91
3571 .sleb128 56
3572 .uleb128 0x1c
3573 .4 byte $LC253
3574 .byte 0x1
3575 .byte 0x1d
3576 .4 byte 0x78
3577 .byte 0x2
3578 .byte 0x91
3579 .sleb128 60
3580 .uleb128 0x1e
3581 .4 byte $LBB4
3582 .4 byte $LBE4

```

```

3583 .uleb128 0x1f
3584 .ascii "im\000"
3585 .byte 0x1
3586 .byte 0x1e
3587 .4byte 0x78
3588 .byte 0x2
3589 .byte 0x91
3590 .sleb128 24
3591 .uleb128 0x1e
3592 .4byte $LBB5
3593 .4byte $LBE5
3594 .uleb128 0x1f
3595 .ascii "re\000"
3596 .byte 0x1
3597 .byte 0x1f
3598 .4byte 0x78
3599 .byte 0x2
3600 .byte 0x91
3601 .sleb128 28
3602 .byte 0x0
3603 .byte 0x0
3604 .byte 0x0
3605 .uleb128 0x5
3606 .byte 0x4
3607 .4byte 0xde2
3608 .uleb128 0x5
3609 .byte 0x4
3610 .4byte 0x78
3611 .uleb128 0x1b
3612 .4byte 0xe10
3613 .byte 0x1
3614 .4byte $LC260
3615 .byte 0x1
3616 .byte 0x25
3617 .byte 0x1
3618 .4byte $LFB33
3619 .4byte $LFE33
3620 .4byte $LSFDE4
3621 .byte 0x1
3622 .byte 0x6e
3623 .uleb128 0x20
3624 .ascii "c\000"
3625 .byte 0x1
3626 .byte 0x25
3627 .4byte 0xcb6
3628 .byte 0x2
3629 .byte 0x91
3630 .sleb128 48
3631 .byte 0x0
3632 .uleb128 0x21

```

```

3633 .4 byte 0xe3c
3634 .byte 0x1
3635 .4 byte $LC261
3636 .byte 0x1
3637 .byte 0x29
3638 .byte 0x1
3639 .4 byte 0xbe8
3640 .4 byte $LFB35
3641 .4 byte $LFE35
3642 .4 byte $LSFDE6
3643 .byte 0x1
3644 .byte 0x6e
3645 .uleb128 0x20
3646 .ascii "a\000"
3647 .byte 0x1
3648 .byte 0x29
3649 .4 byte 0xcb6
3650 .byte 0x2
3651 .byte 0x91
3652 .sleb128 48
3653 .byte 0x0
3654 .uleb128 0x1b
3655 .4 byte 0xe72
3656 .byte 0x1
3657 .4 byte $LC262
3658 .byte 0x1
3659 .byte 0x2d
3660 .byte 0x1
3661 .4 byte $LFB37
3662 .4 byte $LFE37
3663 .4 byte $LSFDE8
3664 .byte 0x1
3665 .byte 0x6e
3666 .uleb128 0x20
3667 .ascii "a\000"
3668 .byte 0x1
3669 .byte 0x2d
3670 .4 byte 0xe72
3671 .byte 0x2
3672 .byte 0x91
3673 .sleb128 56
3674 .uleb128 0x1f
3675 .ascii "aux\000"
3676 .byte 0x1
3677 .byte 0x2e
3678 .4 byte 0xbe8
3679 .byte 0x2
3680 .byte 0x91
3681 .sleb128 24
3682 .byte 0x0

```

```

3683 .uleb128 0x5
3684 .byte 0x4
3685 .4byte 0xcb6
3686 .uleb128 0x1b
3687 .4byte 0xea2
3688 .byte 0x1
3689 .4byte $LC263
3690 .byte 0x1
3691 .byte 0x33
3692 .byte 0x1
3693 .4byte $LFB39
3694 .4byte $LFE39
3695 .4byte $LSFDE10
3696 .byte 0x1
3697 .byte 0x6e
3698 .uleb128 0x1c
3699 .4byte $LC264
3700 .byte 0x1
3701 .byte 0x33
3702 .4byte 0x78
3703 .byte 0x2
3704 .byte 0x91
3705 .sleb128 48
3706 .byte 0x0
3707 .uleb128 0x21
3708 .4byte 0xf10
3709 .byte 0x1
3710 .4byte $LC265
3711 .byte 0x1
3712 .byte 0x44
3713 .byte 0x1
3714 .4byte 0xddc
3715 .4byte $LFB41
3716 .4byte $LFE41
3717 .4byte $LSFDE12
3718 .byte 0x1
3719 .byte 0x6e
3720 .uleb128 0x1c
3721 .4byte $LC252
3722 .byte 0x1
3723 .byte 0x44
3724 .4byte 0x78
3725 .byte 0x2
3726 .byte 0x91
3727 .sleb128 48
3728 .uleb128 0x1c
3729 .4byte $LC253
3730 .byte 0x1
3731 .byte 0x44
3732 .4byte 0x78

```

```

3733 .byte 0x2
3734 .byte 0x91
3735 .sleb128 52
3736 .uleb128 0x1c
3737 .4byte $LC264
3738 .byte 0x1
3739 .byte 0x44
3740 .4byte 0x78
3741 .byte 0x2
3742 .byte 0x91
3743 .sleb128 56
3744 .uleb128 0x1d
3745 .4byte $LC259
3746 .byte 0x1
3747 .byte 0x45
3748 .4byte 0xddc
3749 .byte 0x2
3750 .byte 0x91
3751 .sleb128 24
3752 .uleb128 0x1e
3753 .4byte $LBB8
3754 .4byte $LBE8
3755 .uleb128 0x1f
3756 .ascii "i\000"
3757 .byte 0x1
3758 .byte 0x4b
3759 .4byte 0x78
3760 .byte 0x2
3761 .byte 0x91
3762 .sleb128 28
3763 .byte 0x0
3764 .byte 0x0
3765 .uleb128 0x1b
3766 .4byte 0x104b
3767 .byte 0x1
3768 .4byte $LC266
3769 .byte 0x1
3770 .byte 0x55
3771 .byte 0x1
3772 .4byte $LFB43
3773 .4byte $LFE43
3774 .4byte $LSFDE14
3775 .byte 0x1
3776 .byte 0x6e
3777 .uleb128 0x1c
3778 .4byte $LC259
3779 .byte 0x1
3780 .byte 0x55
3781 .4byte 0xddc
3782 .byte 0x3

```



```

3783 .byte 0x91
3784 .sleb128 136
3785 .uleb128 0x1c
3786 .4byte $LC253
3787 .byte 0x1
3788 .byte 0x55
3789 .4byte 0x78
3790 .byte 0x3
3791 .byte 0x91
3792 .sleb128 140
3793 .uleb128 0x1c
3794 .4byte $LC252
3795 .byte 0x1
3796 .byte 0x55
3797 .4byte 0x78
3798 .byte 0x3
3799 .byte 0x91
3800 .sleb128 144
3801 .uleb128 0x20
3802 .ascii "w\000"
3803 .byte 0x1
3804 .byte 0x55
3805 .4byte 0xbe8
3806 .byte 0x3
3807 .byte 0x91
3808 .sleb128 152
3809 .uleb128 0x20
3810 .ascii "H\000"
3811 .byte 0x1
3812 .byte 0x55
3813 .4byte 0xbe8
3814 .byte 0x3
3815 .byte 0x91
3816 .sleb128 160
3817 .uleb128 0x1c
3818 .4byte $LC267
3819 .byte 0x1
3820 .byte 0x55
3821 .4byte 0xcb6
3822 .byte 0x3
3823 .byte 0x91
3824 .sleb128 168
3825 .uleb128 0x1c
3826 .4byte $LC268
3827 .byte 0x1
3828 .byte 0x55
3829 .4byte 0xcb6
3830 .byte 0x3
3831 .byte 0x91
3832 .sleb128 184

```

```

3833 .uleb128 0x1d
3834 .4byte $LC269
3835 .byte 0x1
3836 .byte 0x56
3837 .4byte 0xbe8
3838 .byte 0x2
3839 .byte 0x91
3840 .sleb128 24
3841 .uleb128 0x1f
3842 .ascii "n\000"
3843 .byte 0x1
3844 .byte 0x57
3845 .4byte 0x78
3846 .byte 0x2
3847 .byte 0x91
3848 .sleb128 32
3849 .uleb128 0x1d
3850 .4byte $LC270
3851 .byte 0x1
3852 .byte 0x58
3853 .4byte 0xbe8
3854 .byte 0x2
3855 .byte 0x91
3856 .sleb128 40
3857 .uleb128 0x1d
3858 .4byte $LC271
3859 .byte 0x1
3860 .byte 0x59
3861 .4byte 0xbe8
3862 .byte 0x2
3863 .byte 0x91
3864 .sleb128 48
3865 .uleb128 0x1d
3866 .4byte $LC272
3867 .byte 0x1
3868 .byte 0x5a
3869 .4byte 0xbe8
3870 .byte 0x2
3871 .byte 0x91
3872 .sleb128 56
3873 .uleb128 0x1d
3874 .4byte $LC273
3875 .byte 0x1
3876 .byte 0x5b
3877 .4byte 0xbe8
3878 .byte 0x3
3879 .byte 0x91
3880 .sleb128 64
3881 .uleb128 0x1d
3882 .4byte $LC274

```

```

3883 .byte 0x1
3884 .byte 0x5c
3885 .4byte 0xbe8
3886 .byte 0x3
3887 .byte 0x91
3888 .sleb128 72
3889 .uleb128 0x1d
3890 .4byte $LC275
3891 .byte 0x1
3892 .byte 0x5c
3893 .4byte 0xbe8
3894 .byte 0x3
3895 .byte 0x91
3896 .sleb128 80
3897 .uleb128 0x1e
3898 .4byte $LBB10
3899 .4byte $LBE10
3900 .uleb128 0x1f
3901 .ascii "im\000"
3902 .byte 0x1
3903 .byte 0x6c
3904 .4byte 0x78
3905 .byte 0x3
3906 .byte 0x91
3907 .sleb128 88
3908 .uleb128 0x1e
3909 .4byte $LBB11
3910 .4byte $LBE11
3911 .uleb128 0x1d
3912 .4byte $LC276
3913 .byte 0x1
3914 .byte 0x6e
3915 .4byte 0xcb6
3916 .byte 0x3
3917 .byte 0x91
3918 .sleb128 96
3919 .uleb128 0x1e
3920 .4byte $LBB12
3921 .4byte $LBE12
3922 .uleb128 0x1f
3923 .ascii "re\000"
3924 .byte 0x1
3925 .byte 0x6f
3926 .4byte 0x78
3927 .byte 0x3
3928 .byte 0x91
3929 .sleb128 112
3930 .byte 0x0
3931 .byte 0x0
3932 .byte 0x0

```

3933 .byte 0x0
3934 .uleb128 0x21
3935 .4byte 0x1142
3936 .byte 0x1
3937 .4byte \$LC277
3938 .byte 0x1
3939 .byte 0x81
3940 .byte 0x1
3941 .4byte 0x78
3942 .4byte \$LFB45
3943 .4byte \$LFE45
3944 .4byte \$LSFDE16
3945 .byte 0x1
3946 .byte 0x6e
3947 .uleb128 0x1c
3948 .4byte \$LC278
3949 .byte 0x1
3950 .byte 0x80
3951 .4byte 0x78
3952 .byte 0x3
3953 .byte 0x91
3954 .sleb128 176
3955 .uleb128 0x1c
3956 .4byte \$LC279
3957 .byte 0x1
3958 .byte 0x80
3959 .4byte 0x1142
3960 .byte 0x3
3961 .byte 0x91
3962 .sleb128 180
3963 .uleb128 0x1d
3964 .4byte \$LC264
3965 .byte 0x1
3966 .byte 0x82
3967 .4byte 0x78
3968 .byte 0x3
3969 .byte 0x91
3970 .sleb128 72
3971 .uleb128 0x1d
3972 .4byte \$LC252
3973 .byte 0x1
3974 .byte 0x83
3975 .4byte 0x78
3976 .byte 0x3
3977 .byte 0x91
3978 .sleb128 76
3979 .uleb128 0x1d
3980 .4byte \$LC253
3981 .byte 0x1
3982 .byte 0x84

```

3983 .4byte 0x78
3984 .byte 0x3
3985 .byte 0x91
3986 .sleb128 80
3987 .uleb128 0x1d
3988 .4byte $LC267
3989 .byte 0x1
3990 .byte 0x85
3991 .4byte 0xcb6
3992 .byte 0x3
3993 .byte 0x91
3994 .sleb128 88
3995 .uleb128 0x1d
3996 .4byte $LC268
3997 .byte 0x1
3998 .byte 0x86
3999 .4byte 0xcb6
4000 .byte 0x3
4001 .byte 0x91
4002 .sleb128 104
4003 .uleb128 0x1d
4004 .4byte $LC280
4005 .byte 0x1
4006 .byte 0x8b
4007 .4byte 0xd6
4008 .byte 0x3
4009 .byte 0x91
4010 .sleb128 120
4011 .uleb128 0x1d
4012 .4byte $LC281
4013 .byte 0x1
4014 .byte 0x8c
4015 .4byte 0xd6
4016 .byte 0x3
4017 .byte 0x91
4018 .sleb128 124
4019 .uleb128 0x1f
4020 .ascii "H\000"
4021 .byte 0x1
4022 .byte 0x8d
4023 .4byte 0xbe8
4024 .byte 0x3
4025 .byte 0x91
4026 .sleb128 128
4027 .uleb128 0x1f
4028 .ascii "w\000"
4029 .byte 0x1
4030 .byte 0x8d
4031 .4byte 0xbe8
4032 .byte 0x3

```

```

4033 .byte 0x91
4034 .sleb128 136
4035 .uleb128 0x1d
4036 .4byte $LC251
4037 .byte 0x1
4038 .byte 0x8e
4039 .4byte 0xd34
4040 .byte 0x3
4041 .byte 0x91
4042 .sleb128 144
4043 .uleb128 0x22
4044 .4byte $LC259
4045 .byte 0x1
4046 .2byte 0x10c
4047 .4byte 0xddc
4048 .byte 0x3
4049 .byte 0x91
4050 .sleb128 148
4051 .uleb128 0x1e
4052 .4byte $LBB14
4053 .4byte $LBE14
4054 .uleb128 0x1f
4055 .ascii "i\000"
4056 .byte 0x1
4057 .byte 0x91
4058 .4byte 0x78
4059 .byte 0x3
4060 .byte 0x91
4061 .sleb128 148
4062 .byte 0x0
4063 .byte 0x0
4064 .uleb128 0x5
4065 .byte 0x4
4066 .4byte 0xd6
4067 .uleb128 0x23
4068 .byte 0x1
4069 .4byte 0x3c2
4070 .uleb128 0x24
4071 .4byte $LC282
4072 .byte 0x5
4073 .byte 0x8d
4074 .4byte 0x1148
4075 .byte 0x1
4076 .byte 0x1
4077 .uleb128 0x15
4078 .4byte $LC284
4079 .byte 0x1
4080 .byte 0x0
4081 .section .debug_abbrev
4082 .uleb128 0x1

```

4083 .uleb128 0x11
4084 .byte 0x1
4085 .uleb128 0x10
4086 .uleb128 0x6
4087 .uleb128 0x12
4088 .uleb128 0x1
4089 .uleb128 0x11
4090 .uleb128 0x1
4091 .uleb128 0x3
4092 .uleb128 0xe
4093 .uleb128 0x1b
4094 .uleb128 0xe
4095 .uleb128 0x25
4096 .uleb128 0xe
4097 .uleb128 0x13
4098 .uleb128 0xb
4099 .byte 0x0
4100 .byte 0x0
4101 .uleb128 0x2
4102 .uleb128 0x16
4103 .byte 0x0
4104 .uleb128 0x3
4105 .uleb128 0xe
4106 .uleb128 0x3a
4107 .uleb128 0xb
4108 .uleb128 0x3b
4109 .uleb128 0xb
4110 .uleb128 0x49
4111 .uleb128 0x13
4112 .byte 0x0
4113 .byte 0x0
4114 .uleb128 0x3
4115 .uleb128 0x24
4116 .byte 0x0
4117 .uleb128 0x3
4118 .uleb128 0xe
4119 .uleb128 0xb
4120 .uleb128 0xb
4121 .uleb128 0x3e
4122 .uleb128 0xb
4123 .byte 0x0
4124 .byte 0x0
4125 .uleb128 0x4
4126 .uleb128 0x24
4127 .byte 0x0
4128 .uleb128 0x3
4129 .uleb128 0x8
4130 .uleb128 0xb
4131 .uleb128 0xb
4132 .uleb128 0x3e

4133 .uleb128 0xb
4134 .byte 0x0
4135 .byte 0x0
4136 .uleb128 0x5
4137 .uleb128 0xf
4138 .byte 0x0
4139 .uleb128 0xb
4140 .uleb128 0xb
4141 .uleb128 0x49
4142 .uleb128 0x13
4143 .byte 0x0
4144 .byte 0x0
4145 .uleb128 0x6
4146 .uleb128 0x17
4147 .byte 0x1
4148 .uleb128 0x1
4149 .uleb128 0x13
4150 .uleb128 0xb
4151 .uleb128 0xb
4152 .uleb128 0x3a
4153 .uleb128 0xb
4154 .uleb128 0x3b
4155 .uleb128 0xb
4156 .byte 0x0
4157 .byte 0x0
4158 .uleb128 0x7
4159 .uleb128 0xd
4160 .byte 0x0
4161 .uleb128 0x3
4162 .uleb128 0xe
4163 .uleb128 0x3a
4164 .uleb128 0xb
4165 .uleb128 0x3b
4166 .uleb128 0xb
4167 .uleb128 0x49
4168 .uleb128 0x13
4169 .byte 0x0
4170 .byte 0x0
4171 .uleb128 0x8
4172 .uleb128 0x1
4173 .byte 0x1
4174 .uleb128 0x1
4175 .uleb128 0x13
4176 .uleb128 0x49
4177 .uleb128 0x13
4178 .byte 0x0
4179 .byte 0x0
4180 .uleb128 0x9
4181 .uleb128 0x21
4182 .byte 0x0

4183 .uleb128 0x49
4184 .uleb128 0x13
4185 .uleb128 0x2f
4186 .uleb128 0xb
4187 .byte 0x0
4188 .byte 0x0
4189 .uleb128 0xa
4190 .uleb128 0x13
4191 .byte 0x1
4192 .uleb128 0x1
4193 .uleb128 0x13
4194 .uleb128 0x3
4195 .uleb128 0xe
4196 .uleb128 0xb
4197 .uleb128 0xb
4198 .uleb128 0x3a
4199 .uleb128 0xb
4200 .uleb128 0x3b
4201 .uleb128 0xb
4202 .byte 0x0
4203 .byte 0x0
4204 .uleb128 0xb
4205 .uleb128 0xd
4206 .byte 0x0
4207 .uleb128 0x3
4208 .uleb128 0xe
4209 .uleb128 0x3a
4210 .uleb128 0xb
4211 .uleb128 0x3b
4212 .uleb128 0xb
4213 .uleb128 0x49
4214 .uleb128 0x13
4215 .uleb128 0x38
4216 .uleb128 0xa
4217 .byte 0x0
4218 .byte 0x0
4219 .uleb128 0xc
4220 .uleb128 0xd
4221 .byte 0x0
4222 .uleb128 0x3
4223 .uleb128 0x8
4224 .uleb128 0x3a
4225 .uleb128 0xb
4226 .uleb128 0x3b
4227 .uleb128 0xb
4228 .uleb128 0x49
4229 .uleb128 0x13
4230 .uleb128 0x38
4231 .uleb128 0xa
4232 .byte 0x0

4233 .byte 0x0
4234 .uleb128 0xd
4235 .uleb128 0xf
4236 .byte 0x0
4237 .uleb128 0xb
4238 .uleb128 0xb
4239 .byte 0x0
4240 .byte 0x0
4241 .uleb128 0xe
4242 .uleb128 0x15
4243 .byte 0x1
4244 .uleb128 0x1
4245 .uleb128 0x13
4246 .uleb128 0x27
4247 .uleb128 0xc
4248 .uleb128 0x49
4249 .uleb128 0x13
4250 .byte 0x0
4251 .byte 0x0
4252 .uleb128 0xf
4253 .uleb128 0x5
4254 .byte 0x0
4255 .uleb128 0x49
4256 .uleb128 0x13
4257 .byte 0x0
4258 .byte 0x0
4259 .uleb128 0x10
4260 .uleb128 0x26
4261 .byte 0x0
4262 .uleb128 0x49
4263 .uleb128 0x13
4264 .byte 0x0
4265 .byte 0x0
4266 .uleb128 0x11
4267 .uleb128 0x16
4268 .byte 0x0
4269 .uleb128 0x3
4270 .uleb128 0xe
4271 .uleb128 0x3a
4272 .uleb128 0xb
4273 .uleb128 0x3b
4274 .uleb128 0x5
4275 .uleb128 0x49
4276 .uleb128 0x13
4277 .byte 0x0
4278 .byte 0x0
4279 .uleb128 0x12
4280 .uleb128 0x35
4281 .byte 0x0
4282 .uleb128 0x49

4283 .uleb128 0x13
4284 .byte 0x0
4285 .byte 0x0
4286 .uleb128 0x13
4287 .uleb128 0x13
4288 .byte 0x1
4289 .uleb128 0x1
4290 .uleb128 0x13
4291 .uleb128 0x3
4292 .uleb128 0xe
4293 .uleb128 0xb
4294 .uleb128 0xb
4295 .uleb128 0x3a
4296 .uleb128 0xb
4297 .uleb128 0x3b
4298 .uleb128 0x5
4299 .byte 0x0
4300 .byte 0x0
4301 .uleb128 0x14
4302 .uleb128 0xd
4303 .byte 0x0
4304 .uleb128 0x3
4305 .uleb128 0xe
4306 .uleb128 0x3a
4307 .uleb128 0xb
4308 .uleb128 0x3b
4309 .uleb128 0x5
4310 .uleb128 0x49
4311 .uleb128 0x13
4312 .uleb128 0x38
4313 .uleb128 0xa
4314 .byte 0x0
4315 .byte 0x0
4316 .uleb128 0x15
4317 .uleb128 0x13
4318 .byte 0x0
4319 .uleb128 0x3
4320 .uleb128 0xe
4321 .uleb128 0x3c
4322 .uleb128 0xc
4323 .byte 0x0
4324 .byte 0x0
4325 .uleb128 0x16
4326 .uleb128 0x13
4327 .byte 0x1
4328 .uleb128 0x1
4329 .uleb128 0x13
4330 .uleb128 0xb
4331 .uleb128 0xb
4332 .uleb128 0x3a

4333 .uleb128 0xb
4334 .uleb128 0x3b
4335 .uleb128 0xb
4336 .byte 0x0
4337 .byte 0x0
4338 .uleb128 0x17
4339 .uleb128 0x17
4340 .byte 0x1
4341 .uleb128 0x1
4342 .uleb128 0x13
4343 .uleb128 0x3
4344 .uleb128 0xe
4345 .uleb128 0xb
4346 .uleb128 0xb
4347 .uleb128 0x3a
4348 .uleb128 0xb
4349 .uleb128 0x3b
4350 .uleb128 0xb
4351 .byte 0x0
4352 .byte 0x0
4353 .uleb128 0x18
4354 .uleb128 0x4
4355 .byte 0x1
4356 .uleb128 0x1
4357 .uleb128 0x13
4358 .uleb128 0x3
4359 .uleb128 0xe
4360 .uleb128 0xb
4361 .uleb128 0xb
4362 .uleb128 0x3a
4363 .uleb128 0xb
4364 .uleb128 0x3b
4365 .uleb128 0xb
4366 .byte 0x0
4367 .byte 0x0
4368 .uleb128 0x19
4369 .uleb128 0x28
4370 .byte 0x0
4371 .uleb128 0x3
4372 .uleb128 0xe
4373 .uleb128 0x1c
4374 .uleb128 0xd
4375 .byte 0x0
4376 .byte 0x0
4377 .uleb128 0x1a
4378 .uleb128 0x28
4379 .byte 0x0
4380 .uleb128 0x3
4381 .uleb128 0xe
4382 .uleb128 0x1c

4383 .uleb128 0xb
4384 .byte 0x0
4385 .byte 0x0
4386 .uleb128 0x1b
4387 .uleb128 0x2e
4388 .byte 0x1
4389 .uleb128 0x1
4390 .uleb128 0x13
4391 .uleb128 0x3f
4392 .uleb128 0xc
4393 .uleb128 0x3
4394 .uleb128 0xe
4395 .uleb128 0x3a
4396 .uleb128 0xb
4397 .uleb128 0x3b
4398 .uleb128 0xb
4399 .uleb128 0x27
4400 .uleb128 0xc
4401 .uleb128 0x11
4402 .uleb128 0x1
4403 .uleb128 0x12
4404 .uleb128 0x1
4405 .uleb128 0x2001
4406 .uleb128 0x6
4407 .uleb128 0x40
4408 .uleb128 0xa
4409 .byte 0x0
4410 .byte 0x0
4411 .uleb128 0x1c
4412 .uleb128 0x5
4413 .byte 0x0
4414 .uleb128 0x3
4415 .uleb128 0xe
4416 .uleb128 0x3a
4417 .uleb128 0xb
4418 .uleb128 0x3b
4419 .uleb128 0xb
4420 .uleb128 0x49
4421 .uleb128 0x13
4422 .uleb128 0x2
4423 .uleb128 0xa
4424 .byte 0x0
4425 .byte 0x0
4426 .uleb128 0x1d
4427 .uleb128 0x34
4428 .byte 0x0
4429 .uleb128 0x3
4430 .uleb128 0xe
4431 .uleb128 0x3a
4432 .uleb128 0xb

4433 .uleb128 0x3b
4434 .uleb128 0xb
4435 .uleb128 0x49
4436 .uleb128 0x13
4437 .uleb128 0x2
4438 .uleb128 0xa
4439 .byte 0x0
4440 .byte 0x0
4441 .uleb128 0x1e
4442 .uleb128 0xb
4443 .byte 0x1
4444 .uleb128 0x11
4445 .uleb128 0x1
4446 .uleb128 0x12
4447 .uleb128 0x1
4448 .byte 0x0
4449 .byte 0x0
4450 .uleb128 0x1f
4451 .uleb128 0x34
4452 .byte 0x0
4453 .uleb128 0x3
4454 .uleb128 0x8
4455 .uleb128 0x3a
4456 .uleb128 0xb
4457 .uleb128 0x3b
4458 .uleb128 0xb
4459 .uleb128 0x49
4460 .uleb128 0x13
4461 .uleb128 0x2
4462 .uleb128 0xa
4463 .byte 0x0
4464 .byte 0x0
4465 .uleb128 0x20
4466 .uleb128 0x5
4467 .byte 0x0
4468 .uleb128 0x3
4469 .uleb128 0x8
4470 .uleb128 0x3a
4471 .uleb128 0xb
4472 .uleb128 0x3b
4473 .uleb128 0xb
4474 .uleb128 0x49
4475 .uleb128 0x13
4476 .uleb128 0x2
4477 .uleb128 0xa
4478 .byte 0x0
4479 .byte 0x0
4480 .uleb128 0x21
4481 .uleb128 0x2e
4482 .byte 0x1

4483 .uleb128 0x1
4484 .uleb128 0x13
4485 .uleb128 0x3f
4486 .uleb128 0xc
4487 .uleb128 0x3
4488 .uleb128 0xe
4489 .uleb128 0x3a
4490 .uleb128 0xb
4491 .uleb128 0x3b
4492 .uleb128 0xb
4493 .uleb128 0x27
4494 .uleb128 0xc
4495 .uleb128 0x49
4496 .uleb128 0x13
4497 .uleb128 0x11
4498 .uleb128 0x1
4499 .uleb128 0x12
4500 .uleb128 0x1
4501 .uleb128 0x2001
4502 .uleb128 0x6
4503 .uleb128 0x40
4504 .uleb128 0xa
4505 .byte 0x0
4506 .byte 0x0
4507 .uleb128 0x22
4508 .uleb128 0x34
4509 .byte 0x0
4510 .uleb128 0x3
4511 .uleb128 0xe
4512 .uleb128 0x3a
4513 .uleb128 0xb
4514 .uleb128 0x3b
4515 .uleb128 0x5
4516 .uleb128 0x49
4517 .uleb128 0x13
4518 .uleb128 0x2
4519 .uleb128 0xa
4520 .byte 0x0
4521 .byte 0x0
4522 .uleb128 0x23
4523 .uleb128 0x1
4524 .byte 0x0
4525 .uleb128 0x3c
4526 .uleb128 0xb
4527 .uleb128 0x49
4528 .uleb128 0x13
4529 .byte 0x0
4530 .byte 0x0
4531 .uleb128 0x24
4532 .uleb128 0x34

```

4533 .byte 0x0
4534 .uleb128 0x3
4535 .uleb128 0xe
4536 .uleb128 0x3a
4537 .uleb128 0xb
4538 .uleb128 0x3b
4539 .uleb128 0xb
4540 .uleb128 0x49
4541 .uleb128 0x13
4542 .uleb128 0x3f
4543 .uleb128 0xc
4544 .uleb128 0x3c
4545 .uleb128 0xc
4546 .byte 0x0
4547 .byte 0x0
4548 .byte 0x0
4549 .section .debug_pubnames,"",@progbits
4550 .4 byte 0xa7
4551 .2 byte 0x2
4552 .4 byte $Ldebug_info0
4553 .4 byte 0x1162
4554 .4 byte 0xcc1
4555 .ascii "armar_headerPGM\000"
4556 .4 byte 0xd5a
4557 .ascii "armar_imagenPGM\000"
4558 .4 byte 0xde8
4559 .ascii "imprimir_complejo\000"
4560 .4 byte 0xe10
4561 .ascii "abs_cplx\000"
4562 .4 byte 0xe3c
4563 .ascii "sqr_cplx\000"
4564 .4 byte 0xe78
4565 .ascii "imprimir_error\000"
4566 .4 byte 0xea2
4567 .ascii "create_matrix\000"
4568 .4 byte 0xf10
4569 .ascii "generate_julia\000"
4570 .4 byte 0x104b
4571 .ascii "main\000"
4572 .4 byte 0x0
4573 .section .debug_aranges,"",@progbits
4574 .4 byte 0x1c
4575 .2 byte 0x2
4576 .4 byte $Ldebug_info0
4577 .byte 0x4
4578 .byte 0x0
4579 .2 byte 0x0
4580 .2 byte 0x0
4581 .4 byte $Ltext0
4582 .4 byte $Letext0-$Ltext0

```



```

4583 .4 byte 0x0
4584 .4 byte 0x0
4585 .section .debug_str,"MS",@progbits,1
4586 $LC276:
4587 .ascii "zeta\000"
4588 $LC78:
4589 .ascii "_ext\000"
4590 $LC251:
4591 .ascii "salida\000"
4592 $LC68:
4593 .ascii "_size\000"
4594 $LC149:
4595 .ascii "timer_t\000"
4596 $LC165:
4597 .ascii "pthread_mutex_t\000"
4598 $LC207:
4599 .ascii "__pthread_rwlockattr_st\000"
4600 $LC161:
4601 .ascii "__pthread_attr_st\000"
4602 $LC101:
4603 .ascii "uint32_t\000"
4604 $LC169:
4605 .ascii "ptm_interlock\000"
4606 $LC248:
4607 .ascii "real\000"
4608 $LC168:
4609 .ascii "ptm_lock\000"
4610 $LC255:
4611 .ascii "ancho_str\000"
4612 $LC36:
4613 .ascii "short unsigned int\000"
4614 $LC232:
4615 .ascii "float\000"
4616 $LC72:
4617 .ascii "_lbfsize\000"
4618 $LC156:
4619 .ascii "pthread_queue_t\000"
4620 $LC32:
4621 .ascii "unsigned char\000"
4622 $LC262:
4623 .ascii "sqr_cplx\000"
4624 $LC189:
4625 .ascii "__pthread_once_st\000"
4626 $LC269:
4627 .ascii "aux_im\000"
4628 $LC136:
4629 .ascii "mode_t\000"
4630 $LC151:
4631 .ascii "useconds_t\000"
4632 $LC214:

```

```

4633      .ascii  "ptb_waiters\000"
4634 $LC209:
4635      .ascii  "ptr_private\000"
4636 $LC176:
4637      .ascii  "ptma_private\000"
4638 $LC84:
4639      .ascii  "off_t\000"
4640 $LC231:
4641      .ascii  "__val\000"
4642 $LC203:
4643      .ascii  "ptr_nreaders\000"
4644 $LC205:
4645      .ascii  "ptr_private\000"
4646 $LC66:
4647      .ascii  "__sbuf\000"
4648 $LC76:
4649      .ascii  "_seek\000"
4650 $LC50:
4651      .ascii  "__in_port_t\000"
4652 $LC62:
4653      .ascii  "size_t\000"
4654 $LC122:
4655      .ascii  "longlong_t\000"
4656 $LC267:
4657      .ascii  "constant\000"
4658 $LC133:
4659      .ascii  "id_t\000"
4660 $LC75:
4661      .ascii  "_read\000"
4662 $LC123:
4663      .ascii  "u_longlong_t\000"
4664 $LC52:
4665      .ascii  "__off_t\000"
4666 $LC273:
4667      .ascii  "ymax\000"
4668 $LC199:
4669      .ascii  "ptr_magic\000"
4670 $LC208:
4671      .ascii  "ptr_magic\000"
4672 $LC191:
4673      .ascii  "pto_done\000"
4674 $LC89:
4675      .ascii  "mips_fpreg_t\000"
4676 $LC58:
4677      .ascii  "__fsfilcnt_t\000"
4678 $LC150:
4679      .ascii  "suseconds_t\000"
4680 $LC127:
4681      .ascii  "fsfilcnt_t\000"
4682 $LC118:

```

```

4683      .ascii  "cpuid_t\000"
4684 $LC257:
4685      .ascii  "armar_headerPGM\000"
4686 $LC69:
4687      .ascii  "__sFILE\000"
4688 $LC59:
4689      .ascii  "__mbstate8\000"
4690 $LC265:
4691      .ascii  "create_matrix\000"
4692 $LC87:
4693      .ascii  "mips_ureg_t\000"
4694 $LC157:
4695      .ascii  "ptqh_first\000"
4696 $LC112:
4697      .ascii  "u_int\000"
4698 $LC96:
4699      .ascii  "int8_t\000"
4700 $LC152:
4701      .ascii  "__fd_mask\000"
4702 $LC60:
4703      .ascii  "__mbstateL\000"
4704 $LC195:
4705      .ascii  "pts_spin\000"
4706 $LC256:
4707      .ascii  "concat\000"
4708 $LC114:
4709      .ascii  "unchar\000"
4710 $LC125:
4711      .ascii  "blksize_t\000"
4712 $LC177:
4713      .ascii  "pthread_cond_t\000"
4714 $LC172:
4715      .ascii  "ptm_private\000"
4716 $LC245:
4717      .ascii  "arg1\000"
4718 $LC246:
4719      .ascii  "arg2\000"
4720 $LC175:
4721      .ascii  "ptma_magic\000"
4722 $LC286:
4723      .ascii  "/root/tp\000"
4724 $LC229:
4725      .ascii  "qdiv_t\000"
4726 $LC91:
4727      .ascii  "p_size_t\000"
4728 $LC213:
4729      .ascii  "ptb_lock\000"
4730 $LC74:
4731      .ascii  "_close\000"
4732 $LC202:

```

```

4733      .ascii  "ptr_wblocked\000"
4734 $LC134:
4735      .ascii  "ino_t\000"
4736 $LC254:
4737      .ascii  "alto_str\000"
4738 $LC64:
4739      .ascii  "fpos_t\000"
4740 $LC42:
4741      .ascii  "__uint64_t\000"
4742 $LC278:
4743      .ascii  "argc\000"
4744 $LC138:
4745      .ascii  "pid_t\000"
4746 $LC93:
4747      .ascii  "vsize_t\000"
4748 $LC281:
4749      .ascii  "auxc2\000"
4750 $LC103:
4751      .ascii  "uint64_t\000"
4752 $LC153:
4753      .ascii  "fd_set\000"
4754 $LC279:
4755      .ascii  "argv\000"
4756 $LC73:
4757      .ascii  "_cookie\000"
4758 $LC228:
4759      .ascii  "lldiv_t\000"
4760 $LC88:
4761      .ascii  "long unsigned int\000"
4762 $LC187:
4763      .ascii  "ptca_private\000"
4764 $LC230:
4765      .ascii  "__dummy\000"
4766 $LC282:
4767      .ascii  "__sF\000"
4768 $LC65:
4769      .ascii  "__sfpos\000"
4770 $LC53:
4771      .ascii  "__pid_t\000"
4772 $LC81:
4773      .ascii  "_blksize\000"
4774 $LC238:
4775      .ascii  "fdlibm_ieee\000"
4776 $LC67:
4777      .ascii  "_base\000"
4778 $LC234:
4779      .ascii  "__double_u\000"
4780 $LC135:
4781      .ascii  "key_t\000"
4782 $LC121:

```

```

4783     .ascii  "qaddr_t\000"
4784 $LC38:
4785     .ascii  "__uint32_t\000"
4786 $LC259:
4787     .ascii  "matrix_PGM\000"
4788 $LC210:
4789     .ascii  "pthread_barrier_t\000"
4790 $LC183:
4791     .ascii  "ptc_private\000"
4792 $LC225:
4793     .ascii  "quot\000"
4794 $LC130:
4795     .ascii  "dev_t\000"
4796 $LC236:
4797     .ascii  "__long_double_u\000"
4798 $LC145:
4799     .ascii  "clock_t\000"
4800 $LC139:
4801     .ascii  "lwpid_t\000"
4802 $LC274:
4803     .ascii  "deltaX\000"
4804 $LC275:
4805     .ascii  "deltaY\000"
4806 $LC271:
4807     .ascii  "xmax\000"
4808 $LC242:
4809     .ascii  "exception\000"
4810 $LC79:
4811     .ascii  "_ubuf\000"
4812 $LC80:
4813     .ascii  "_nbuf\000"
4814 $LC285:
4815     .ascii  "tp0.c\000"
4816 $LC34:
4817     .ascii  "short_int\000"
4818 $LC104:
4819     .ascii  "u_int8_t\000"
4820 $LC184:
4821     .ascii  "pthread_condattr_t\000"
4822 $LC102:
4823     .ascii  "int64_t\000"
4824 $LC143:
4825     .ascii  "uid_t\000"
4826 $LC221:
4827     .ascii  "ptba_magic\000"
4828 $LC250:
4829     .ascii  "numcomplex\000"
4830 $LC233:
4831     .ascii  "__float_u\000"
4832 $LC237:

```

4833 .ascii "long double\000"
 4834 \$LC241:
 4835 .ascii "fdlibm_posix\000"
 4836 \$LC41:
 4837 .ascii "long long int\000"
 4838 \$LC220:
 4839 .ascii "__pthread_barrierattr_st\000"
 4840 \$LC206:
 4841 .ascii "pthread_rwlockattr_t\000"
 4842 \$LC131:
 4843 .ascii "fixpt_t\000"
 4844 \$LC258:
 4845 .ascii "armar_imagenPGM\000"
 4846 \$LC129:
 4847 .ascii "daddr_t\000"
 4848 \$LC277:
 4849 .ascii "main\000"
 4850 \$LC43:
 4851 .ascii "long long unsigned int\000"
 4852 \$LC287:
 4853 .ascii "GNU C 3.3.3 (NetBSD nb3 20040520) -g\000"
 4854 \$LC56:
 4855 .ascii "__uid_t\000"
 4856 \$LC83:
 4857 .ascii "FILE\000"
 4858 \$LC35:
 4859 .ascii "__uint16_t\000"
 4860 \$LC137:
 4861 .ascii "nlink_t\000"
 4862 \$LC142:
 4863 .ascii "swblk_t\000"
 4864 \$LC155:
 4865 .ascii "pthread_spin_t\000"
 4866 \$LC141:
 4867 .ascii "segsz_t\000"
 4868 \$LC219:
 4869 .ascii "pthread_barrierattr_t\000"
 4870 \$LC144:
 4871 .ascii "mtime_t\000"
 4872 \$LC47:
 4873 .ascii "char\000"
 4874 \$LC227:
 4875 .ascii "ldiv_t\000"
 4876 \$LC186:
 4877 .ascii "ptca_magic\000"
 4878 \$LC90:
 4879 .ascii "paddr_t\000"
 4880 \$LC212:
 4881 .ascii "ptb_magic\000"
 4882 \$LC216:

```

4883      .ascii  "ptb_curcount\000"
4884 $LC100:
4885      .ascii  "int32_t\000"
4886 $LC188:
4887      .ascii  "pthread_once_t\000"
4888 $LC92:
4889      .ascii  "vaddr_t\000"
4890 $LC222:
4891      .ascii  "ptba_private\000"
4892 $LC163:
4893      .ascii  "pta_flags\000"
4894 $LC97:
4895      .ascii  "uint8_t\000"
4896 $LC280:
4897      .ascii  "auxc\000"
4898 $LC180:
4899      .ascii  "ptc_lock\000"
4900 $LC85:
4901      .ascii  "mips_reg_t\000"
4902 $LC243:
4903      .ascii  "type\000"
4904 $LC201:
4905      .ascii  "ptr_rblocked\000"
4906 $LC128:
4907      .ascii  "caddr_t\000"
4908 $LC268:
4909      .ascii  "center\000"
4910 $LC264:
4911      .ascii  "status\000"
4912 $LC193:
4913      .ascii  "__pthread_spinlock_st\000"
4914 $LC218:
4915      .ascii  "ptb_private\000"
4916 $LC198:
4917      .ascii  "__pthread_rwlock_st\000"
4918 $LC179:
4919      .ascii  "ptc_magic\000"
4920 $LC116:
4921      .ascii  "uint\000"
4922 $LC185:
4923      .ascii  "__pthread_condattr_st\000"
4924 $LC247:
4925      .ascii  "retval\000"
4926 $LC196:
4927      .ascii  "pts_flags\000"
4928 $LC266:
4929      .ascii  "generate_julia\000"
4930 $LC182:
4931      .ascii  "ptc_mutex\000"
4932 $LC283:

```

4933 .ascii "__pthread_st\000"
 4934 \$LC70:
 4935 .ascii "_flags\000"
 4936 \$LC197:
 4937 .ascii "pthread_rwlock_t\000"
 4938 \$LC253:
 4939 .ascii "ancho\000"
 4940 \$LC44:
 4941 .ascii "__intptr_t\000"
 4942 \$LC63:
 4943 .ascii "_pos\000"
 4944 \$LC98:
 4945 .ascii "int16_t\000"
 4946 \$LC124:
 4947 .ascii "blkcnt_t\000"
 4948 \$LC224:
 4949 .ascii "wchar_t\000"
 4950 \$LC190:
 4951 .ascii "pto_mutex\000"
 4952 \$LC240:
 4953 .ascii "fdlibm_xopen\000"
 4954 \$LC192:
 4955 .ascii "pthread_spinlock_t\000"
 4956 \$LC226:
 4957 .ascii "div_t\000"
 4958 \$LC164:
 4959 .ascii "pta_private\000"
 4960 \$LC31:
 4961 .ascii "signed char\000"
 4962 \$LC244:
 4963 .ascii "name\000"
 4964 \$LC166:
 4965 .ascii "__pthread_mutex_st\000"
 4966 \$LC249:
 4967 .ascii "imag\000"
 4968 \$LC51:
 4969 .ascii "__mode_t\000"
 4970 \$LC94:
 4971 .ascii "register_t\000"
 4972 \$LC171:
 4973 .ascii "ptm_blocked\000"
 4974 \$LC288:
 4975 .ascii "fdversion\000"
 4976 \$LC148:
 4977 .ascii "clockid_t\000"
 4978 \$LC217:
 4979 .ascii "ptb_generation\000"
 4980 \$LC39:
 4981 .ascii "unsigned int\000"
 4982 \$LC120:

4983 .ascii "quad_t\000"
 4984 \$LC113:
 4985 .ascii "u_long\000"
 4986 \$LC204:
 4987 .ascii "ptr_writer\000"
 4988 \$LC263:
 4989 .ascii "imprimir_error\000"
 4990 \$LC200:
 4991 .ascii "ptr_interlock\000"
 4992 \$LC107:
 4993 .ascii "u_int64_t\000"
 4994 \$LC261:
 4995 .ascii "abs_cplx\000"
 4996 \$LC173:
 4997 .ascii "pthread_mutexattr_t\000"
 4998 \$LC170:
 4999 .ascii "ptm_owner\000"
 5000 \$LC223:
 5001 .ascii "pthread_key_t\000"
 5002 \$LC260:
 5003 .ascii "imprimir_complejo\000"
 5004 \$LC117:
 5005 .ascii "ulong\000"
 5006 \$LC110:
 5007 .ascii "u_char\000"
 5008 \$LC55:
 5009 .ascii "__socklen_t\000"
 5010 \$LC146:
 5011 .ascii "ssize_t\000"
 5012 \$LC178:
 5013 .ascii "__pthread_cond_st\000"
 5014 \$LC40:
 5015 .ascii "__int64_t\000"
 5016 \$LC159:
 5017 .ascii "pthread_t\000"
 5018 \$LC71:
 5019 .ascii "_file\000"
 5020 \$LC108:
 5021 .ascii "in_addr_t\000"
 5022 \$LC29:
 5023 .ascii "__int8_t\000"
 5024 \$LC57:
 5025 .ascii "__fsblkcnt_t\000"
 5026 \$LC132:
 5027 .ascii "gid_t\000"
 5028 \$LC126:
 5029 .ascii "fsblkcnt_t\000"
 5030 \$LC95:
 5031 .ascii "__cpu_simple_lock_t\000"
 5032 \$LC106:

5033 .ascii "uint32_t\000"
5034 \$LC82:
5035 .ascii "_offset\000"
5036 \$LC252:
5037 .ascii "alto\000"
5038 \$LC61:
5039 .ascii "__mbstate_t\000"
5040 \$LC158:
5041 .ascii "ptqh_last\000"
5042 \$LC115:
5043 .ascii "ushort\000"
5044 \$LC45:
5045 .ascii "__uintptr_t\000"
5046 \$LC48:
5047 .ascii "__gid_t\000"
5048 \$LC239:
5049 .ascii "fdlibm_svid\000"
5050 \$LC37:
5051 .ascii "__int32_t\000"
5052 \$LC215:
5053 .ascii "ptb_initcount\000"
5054 \$LC174:
5055 .ascii "__pthread_mutexattr_st\000"
5056 \$LC77:
5057 .ascii "_write\000"
5058 \$LC30:
5059 .ascii "__uint8_t\000"
5060 \$LC154:
5061 .ascii "fds_bits\000"
5062 \$LC162:
5063 .ascii "pta_magic\000"
5064 \$LC284:
5065 .ascii "__pthread_spin_st\000"
5066 \$LC99:
5067 .ascii "uint16_t\000"
5068 \$LC270:
5069 .ascii "xmin\000"
5070 \$LC86:
5071 .ascii "long_int\000"
5072 \$LC160:
5073 .ascii "pthread_attr_t\000"
5074 \$LC46:
5075 .ascii "__caddr_t\000"
5076 \$LC109:
5077 .ascii "in_port_t\000"
5078 \$LC211:
5079 .ascii "__pthread_barrier_st\000"
5080 \$LC105:
5081 .ascii "uint16_t\000"
5082 \$LC235:

```

5083      .ascii  "double\000"
5084 $LC147:
5085      .ascii  "time_t\000"
5086 $LC167:
5087      .ascii  "ptm_magic\000"
5088 $LC119:
5089      .ascii  "u_quad_t\000"
5090 $LC272:
5091      .ascii  "ymin\000"
5092 $LC194:
5093      .ascii  "pts_magic\000"
5094 $LC49:
5095      .ascii  "__in_addr_t\000"
5096 $LC140:
5097      .ascii  "rlim_t\000"
5098 $LC54:
5099      .ascii  "__sa_family_t\000"
5100 $LC111:
5101      .ascii  "u_short\000"
5102 $LC33:
5103      .ascii  "__int16_t\000"
5104 $LC181:
5105      .ascii  "ptc_waiters\000"
5106      .ident   "GCC: (GNU) 3.3.3 (NetBSD nb3 20040520)"
                                tp0.s

```

7. Conclusión

En el presente trabajo se aprendió a utilizar las herramientas que utilizaremos en los próximos trabajos prácticos. Se aprendió a utilizar el emulador GXEmul y compilar programas para arquitectura MIPS, tanto a forma binaria como assembly MIPS.

8. Enunciado

Adjunto a partir de la siguiente hoja.

Univesidad de Buenos Aires - FIUBA
66:20 Organización de Computadoras
Trabajo práctico 0: Infraestructura básica
2º cuatrimestre de 2016

\$Date: 2016/09/20 17:31:41 \$

1. Objetivos

Familiarizarse con las herramientas de software que usaremos en los siguientes trabajos, implementando un programa y su correspondiente documentación que resuelvan el problema descripto más abajo.

2. Alcance

Este trabajo práctico es de elaboración grupal, evaluación individual, y de carácter obligatorio para todos alumnos del curso.

3. Requisitos

El trabajo deberá ser entregado personalmente, en la fecha estipulada, con una carátula que contenga los datos completos de todos los integrantes, un informe impreso de acuerdo con lo que mencionaremos en la sección 6, y con una copia digital de los archivos fuente necesarios para compilar el trabajo.

4. Recursos

Usaremos el programa GXemul [1] para simular el entorno de desarrollo que utilizaremos en este y otros trabajos prácticos, una máquina MIPS corriendo una versión reciente del sistema operativo NetBSD [2].

Durante la primera clase del curso presentaremos brevemente los pasos necesarios para la instalación y configuración del entorno de desarrollo.

5. Programa

Se trata de diseñar un programa que permita dibujar el conjunto de Julia [3] y sus vecindades, en lenguaje C, correspondiente a un polinomio cuadrático.

El mismo recibirá, por línea de comando, una serie de parámetros describiendo la región del plano complejo, las características del archivo imagen a generar, y el parámetro c .

No deberá interactuar con el usuario, ya que no se trata de un programa interactivo, sino más bien de una herramienta de procesamiento *batch*. Al finalizar la ejecución, y volver al sistema operativo, el programa habrá dibujado el fractal en el archivo de salida.

El formato gráfico a usar es PGM o *portable gray map* [4], un formato simple para describir imágenes digitales monocromáticas.

5.1. Algoritmo

El algoritmo básico es simple: para algunos puntos z de la región del plano que estamos procesando haremos un cálculo repetitivo. Terminado el cálculo, asignamos el nivel de intensidad del pixel en base a la condición de corte de ese cálculo.

El color de cada punto representa la “velocidad de escape” asociada con ese número complejo: blanco para aquellos puntos que pertenecen al conjunto (y por ende la “cuenta” permanece acotada), y tonos gradualmente más oscuros para los puntos divergentes, que no pertenezcan al conjunto.

Más específicamente: para cada pixel de la pantalla, tomaremos su punto medio, expresado en coordenadas complejas, $z = z_{re} + z_{im}i$. A continuación, iteramos sobre $z_{n+1} = z_n^2 + c$, con $z_0 = z$. Cortamos la iteración cuando $|z_n| > 2$, o después de N iteraciones.

En pseudo código:

```
para cada pixel $p {
    $z = complejo asociado a $p;
    for ($i = 0; $i < $N - 1; ++$i) {
        if (abs($z) > 2)
            break;
        $z = $z * $z + $c;
    }
    dibujar el punto p con brillo $i;
}
```

Notar que c es un parámetro del programa.

Así tendremos, al finalizar, una representación visual de la cantidad de ciclos de cómputo realizados hasta alcanzar la condición de escape (ver figura 1).

5.2. Interfaz

A fin de facilitar el intercambio de código *ad-hoc*, normalizaremos algunas de las opciones que deberán ser provistas por el programa:

- **-r**: permite cambiar la resolución de la imagen generada. El valor por defecto será de 640x480 puntos.
- **-c**: para especificar el centro de la imagen, el punto central de la porción del plano complejo dibujada, expresado en forma binómica (i.e. $a + bi$). Por defecto usaremos $0 + 0i$.
- **-C**: determina el parámetro c , también expresado en forma binómica. El valor por defecto será $0,285 - 0,01i$.
- **-w**: especifica el ancho del rectángulo que contiene la región del plano complejo que estamos por dibujar. Valor por defecto: 4.

- `-H`: sirve, en forma similar, para especificar el alto del rectángulo a dibujar. Valor por defecto: 4.
- `-o`: permite colocar la imagen de salida, (en formato PGM [4]) en el archivo pasado como argumento; o por salida estándar `-stdout` si el argumento es “-”.

5.3. Casos de prueba

Es necesario que el informe trabajo práctico incluya una sección dedicada a verificar el funcionamiento del código implementado.

En el caso del TP 0, será necesario escribir pruebas orientadas a probar el programa completo, ejercitando los casos más comunes de funcionamiento, los casos de borde, y también casos de error.

Incluimos en el apéndice A algunos ejemplos de casos de interés, orientados a ejercitar algunos errores y condiciones de borde.

5.4. Ejemplos

Generamos un dibujo usando los valores por defecto, barriendo la región rectangular del plano comprendida entre los vértices $-2 + 2i$ y $+2 - 2i$.

```
$ tp0 -o uno.pgm
```

La figura 1 muestra la imagen `uno.pgm`.

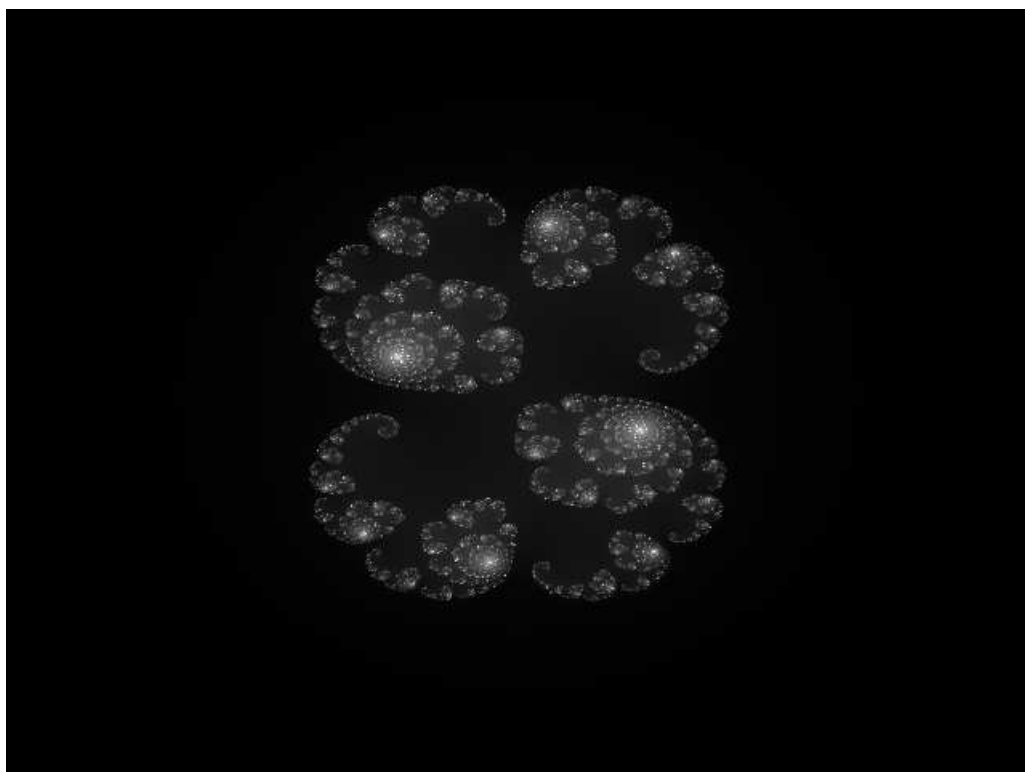


Figura 1: Región barrida por defecto.

A continuación, hacemos *zoom* sobre la región centrada en $+0,282 - 0,01i$, usando un rectángulo de 0,005 unidades de lado. El resultado podemos observarlo en la figura 2.

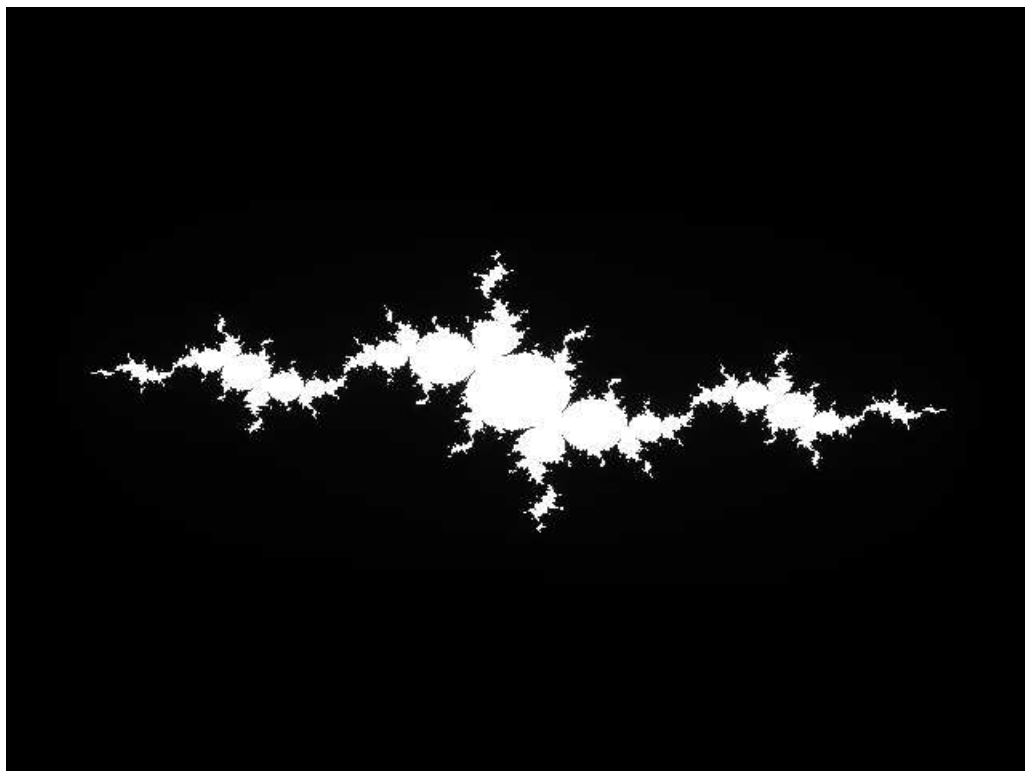


Figura 2: Región por default con $C = -1,125 - 0,21650635094611i$.

```
$ tp0 -C -1.125-0.21650635094611i -o dos.pgm
```

6. Informe

El informe deberá incluir:

- Documentación relevante al diseño e implementación del programa.
- Documentación relevante al proceso de compilación: cómo obtener el ejecutable a partir de los archivos fuente.
- Las corridas de prueba, con los comentarios pertinentes.
- El código fuente, en lenguaje C.
- El código MIPS32 generado por el compilador¹;
- Este enunciado.

7. Fechas

Fecha de vencimiento: Martes 26/9.

¹Por motivos prácticos, en la copia impresa sólo es necesario incluir la primera página del código assembly MIPS32 generado por el compilador.

Referencias

- [1] GXemul, <http://gavare.se/gxemul/>.
- [2] The NetBSD project.
<http://www.netbsd.org/>.
- [3] http://en.wikipedia.org/wiki/Julia_set (Wikipedia).
- [4] PGM format specification.
<http://netpbm.sourceforge.net/doc/pgm.html>.

A. Algunos casos de prueba

1. Generamos una imagen de 1 punto de lado, centrada en el origen del plano complejo:

```
$ tp0 -c 0.01+0i -r 1x1 -o -  
P2  
1  
1  
255  
255
```

Notar que el resultado es correcto, ya que este punto pertenece al conjunto de Julia.

2. Repetimos el experimento, pero nos centramos ahora en un punto que *seguro* no pertenece al conjunto:

```
$ tp0 -c 10+0i -r 1x1 -o -  
P2  
1  
1  
255  
0
```

Notar que el resultado es correcto, ya que este punto no pertenece al conjunto de Julia.

3. Imagen imposible:

```
$ tp0 -c 0+0i -r 0x1 -o -  
Usage:  
  tp0 -h  
  tp0 -V  
...
```

4. Archivo de salida imposible:

```
$ tp0 -o /tmp  
fatal: cannot open output file.
```

5. Coordenadas complejas imposibles:

```
$ tp0 -c 1+3 -o -  
fatal: invalid center specification.
```

6. Argumentos de línea de comando vacíos,

```
$ tp0 -c "" -o -  
fatal: invalid center specification.
```