

LAB-6 REPORT

3200105787 Yunce Zhang

LAB TARGET

Write a program which can turn LC-3 assembly code to binary machine code.

CODE:

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <math.h>
4
5  void dectobin16(int n);
6  void dectobin4(int n);
7  void dectobin3(int n);
8  void dectobin5(int n);
9  void dectobin9(int n);
10 void dectobin6(int n);
11 void dectobin8(int n);
12 void dectobin11(int n);
13 void signeddectobin5(int n);
14 void signeddectobin6(int n);
15 void signeddectobin16(int n);
16 void signeddectobin9(int n);
17 void signeddectobin11(int n);
18
19
20
21
22 int main (void){
23 // char str0[100];
24 // char ch;
25 // int button;
26 // int site=0;
27 int i;
28 int words = 0; /*the number of the words */
29 int end = 1; /*to decide whether it is ended*/
30 int num = 1; /*the number of the number of beginning */
31 int begin = 0; /* the site of the beginning */
32 int beg[4]; /* store the site of the beginning */
33 int opcode=0; /* store what the opcode is*/
34 int out=0; /* used to input the opcode in bin*/
35 int j=0; /*used in the circles*/
36 int p=0; /*用于计算offset (避免.BLKW及.STRINGZ导致的汇编码和机械码行数的区别)
*/
37 int lines=1; /*记录当前行数*/
38 int finallines=0; /*记录总行数 */
39 int l=0; /*label数组中记录标签的编号*/
40 int lin[500]={1}; /*记录每一行末尾的坐标*/
41 char label[100][20]; /*记录每行开头的标签*/
42 int label2[100]={};
```

```

43 // int width; /*used in the instructions that begins with .STRINGZ to
    count the length of the strings*/
44
45 // read the input and count how many words there are at the same time
46     char str[10000][30]={};
47     while(end!=0){
48         scanf ("%s",str[words]);
49         end=strcmp(str[words],".END");
50         words++;
51     }
52 // for(j=0;j<=99;j++){
53 //     for(i=0;i<=14;i++){
54 //         ch=getchar();
55 //         printf("%d\n",ch);
56 //         printf("%c\n",ch);
57 //         if((ch!='\n'&&ch!=32&&ch!=9)){
58 //             str[j][i]=ch;
59 //             button=1;
60 //         }
61
62 //         else if((ch=='\n' || ch==32 || ch==9)&&button==1){
63 //             str[j][i]='\0';
64 //             i=15;
65 //             words++;
66 //             button=0;
67 //         }
68 //         else if((ch=='\n' || ch==32 || ch==9)&&button==0){
69 //             i=-1;
70 //         }
71 //     }
72 //     if(strcmp(str[j],".END")==0){
73 //         j=100;
74 //     }
75 // }
76 // words--;
77 //// 输出来检查
78 // for(i=0;i<words;i++){
79 //     printf ("str[%d]=%s \n",i,str[i]);
80 // }
81 // 读入起始位置
82     i=0;
83 //     printf("%d\n",str[1][num]-48);
84 //     printf("%c\n",str[1][num]);
85     while ((str[1][num]-48>=0 &&str[1][num]-48<=9) || (str[1][num]-65>=0
&&str[1][num]-65<=5)){
86 //         printf("%d \n",str[1][num]-48);
87         if ((str[1][num]-48>=0 &&str[1][num]-48<=9))    beg[i]=str[1]
[num]-48;
88         if ((str[1][num]-65>=0 &&str[1][num]-65<=5))    beg[i]=str[1]
[num]-55;
89         i++;
90         num++;
91     }
92     num--;
93 //     printf ("%d",num);
94
95 //计算出起始位置的十进制表示
96     for (i=4-num;i<num;i++){

```

```

97         begin+=beg[3-i]*pow(16,i);
98     }
99     // printf ("%d",begin);
100    //转换为十六位二进制输出
101    dectobin16(begin);
102    printf ("\n");
103
104    //labels
105    for (i=2;i<words;i++){
106        if (strcmp(str[i],"ADD")==0){
107            i+=3;
108            lines++;
109            lin[lines]=i;
110        }
111        else if (strcmp(str[i],"AND")==0){
112            i+=3;
113            lines++;
114            lin[lines]=i;
115        }
116        else if (strcmp(str[i],"NOT")==0){
117            i+=2;
118            lines++;
119            lin[lines]=i;
120        }
121        else if (strcmp(str[i],"LD")==0){
122            i+=2;
123            lines++;
124            lin[lines]=i;
125        }
126        else if (strcmp(str[i],"LDR")==0){
127            i+=3;
128            lines++;
129            lin[lines]=i;
130        }
131        else if (strcmp(str[i],"LDI")==0){
132            i+=2;
133            lines++;
134            lin[lines]=i;
135        }
136        else if (strcmp(str[i],"LEA")==0){
137            i+=2;
138            lines++;
139            lin[lines]=i;
140        }
141        else if (strcmp(str[i],"ST")==0){
142            i+=2;
143            lines++;
144            lin[lines]=i;
145        }
146        else if (strcmp(str[i],"STR")==0){
147            i+=3;
148            lines++;
149            lin[lines]=i;
150        }
151        else if (strcmp(str[i],"STI")==0){
152            i+=2;
153            lines++;
154            lin[lines]=i;

```

```

155     }
156     else if (strcmp(str[i], "TRAP")==0){
157         i+=1;
158         lines++;
159         lin[lines]=i;
160     }
161     else if (strcmp(str[i], "HALT")==0){
162         lines++;
163         lin[lines]=i;
164     }
165     else if (strcmp(str[i], "JMP")==0){
166         i+=1;
167         lines++;
168         lin[lines]=i;
169     }
170     else if (strcmp(str[i], "RET")==0){
171         lines++;
172         lin[lines]=i;
173     }
174     else if (strcmp(str[i], "JSR")==0){
175         i+=1;
176         lines++;
177         lin[lines]=i;
178     }
179     else if (strcmp(str[i], "JSRR")==0){
180         i+=1;
181         lines++;
182         lin[lines]=i;
183     }
184     else if (strcmp(str[i], "RTI")==0){
185         lines++;
186         lin[lines]=i;
187     }
188     else if (strcmp(str[i], ".FILL")==0){
189         i+=1;
190         lines++;
191         lin[lines]=i;
192     }
193     else if (strcmp(str[i], ".BLKW")==0){
194         i+=1;
195         j=1;
196         out=0;
197         while(str[i][j]!=0){
198             out=out*10+(str[i][j]-48);
199             j++;
200         }
201         for(j=0; j<out; j++){
202             lines++;
203             lin[lines]=i;
204         }
205     }
206     else if (strcmp(str[i], ".STRINGZ")==0){
207         opcode=20;
208         p=0;
209         i++;
210         j=1;
211         while(str[i][j]!=''){
212             if(str[i][j]!=0){

```

```

213         out=str[i][j];
214         j++;
215         p++;
216     }
217     else {
218         i++;
219         p++;
220         j=0;
221     }
222 }
223 for(j=0;j<=p;j++){
224     lines++;
225     lin [lines]=i;
226 }
227
228 }
229 else if (strcmp(str[i],"GETC")==0){
230     lines++;
231     lin [lines]=i;
232 }
233 else if (strcmp(str[i],"OUT")==0){
234     lines++;
235     lin [lines]=i;
236 }
237 else if (strcmp(str[i],"PUTS")==0){
238     lines++;
239     lin [lines]=i;
240 }
241 else if (strcmp(str[i],"IN")==0){
242     lines++;
243     lin [lines]=i;
244 }
245 else if (strcmp(str[i],"PUTSP")==0){
246     lines++;
247     lin [lines]=i;
248 }
249 else if (strcmp(str[i],".END")==0){
250     lines++;
251     lin [lines]=i;
252 }
253 else if
(strcmp(str[i],"BRn")==0||strcmp(str[i],"BRnz")==0||strcmp(str[i],"BRnp")=
=0||strcmp(str[i],"BRnzp")==0||strcmp(str[i],"BRz")==0||strcmp(str[i],"BRz
p")==0||strcmp(str[i],"BRp")==0||strcmp(str[i],"BR")==0){
254     i+=1;
255     lines++;
256     lin[lines]=i;
257 }
258 else {
259     for(j=0;j<=19;j++){
260         label1[j]=str[i][j];
261     }
262     label2[1]=lines;
263     l++;
264 }
265 }
266 finallines=lines;
267 lines=1;

```

```

268 //      for(i=0;i<1;i++){
269 //          printf("1in[%d]=%s %d\n",i,label[i],label2[i]+1);
270 //      }
271
272 //opcodes
273     for (i=2;i<words;i++){
274         if (strcmp(str[i],"ADD")==0){
275             opcode=1;
276             out=1;
277             dectobin4(out);
278             i++;
279             out=str[i][1]-48;
280             dectobin3(out);
281             i++;
282             out=str[i][1]-48;
283             dectobin3(out);
284             i++;
285             if(str[i][0]=='R'){
286                 printf("000");
287                 out=str[i][1]-48;
288                 dectobin3(out);
289                 printf("\n");
290             }
291             if(str[i][0]=='#'){
292                 printf("1");
293                 if(str[i][1]=='-'){
294                     out= str[i][2]-48;
295                     if(str[i][3]!=0)
296                         out=out*10+str[i][3]-48;
297 //                 printf("%d",out);
298                     signeddectobin5(out);
299                     printf("\n");
300                 }
301                 else{
302                     out= str[i][1]-48;
303                     if(str[i][2]!=0)
304                         out=out*10+str[i][2]-48;
305                     dectobin5(out);
306                     printf("\n");
307                 }
308             }
309             if ((str[i][0]=='x' || str[i][0]=='x') && (str[i][1] != 0)){
310                 printf("1");
311                 if(str[i][1]=='-'){
312                     if ((str[i][2]-48>=0 && str[i][2]-48<=9))    out=str[i]
[2]-48;
313                     else if ((str[i][2]-65>=0 && str[i][2]-65<=5))
out=str[i][2]-55;
314                     if(str[i][3]!=0){
315                         if ((str[i][3]-48>=0 && str[i][3]-48<=9))
out=out*16+str[i][3]-48;
316                         else if ((str[i][3]-65>=0 && str[i][3]-65<=5))
out=out*16+str[i][3]-55;
317                     }
318                     signeddectobin5(out);
319                     printf("\n");
320                 }
321                 else{

```

```

322         if ((str[i][1]-48>=0 &&str[i][1]-48<=9)) out=str[i]
[1]-48;
323         else if ((str[i][1]-65>=0 &&str[i][1]-65<=5))
out=str[i][1]-55;
324         if(str[i][2]!=0){
325             if ((str[i][2]-48>=0 &&str[i][2]-48<=9))
out=out*16+str[i][2]-48;
326             else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=out*16+str[i][2]-55;
327         }
328         dectobin5(out);
329         printf("\n");
330     }
331 }
332 lines++;
333
334 }
335 else if (strcmp(str[i],"AND")==0){
336     opcode=2;
337     out=5;
338     dectobin4(out);
339     i++;
340     out=str[i][1]-48;
341     dectobin3(out);
342     i++;
343     out=str[i][1]-48;
344     dectobin3(out);
345     i++;
346     if(str[i][0]=='R'){
347         printf("000");
348         out=str[i][1]-48;
349         dectobin3(out);
350         printf("\n");
351     }
352     if(str[i][0]=='#'){
353         printf("1");
354         if(str[i][1]=='-'){
355             out= str[i][2]-48;
356             if(str[i][3]!=0)
357                 out=out*10+str[i][3]-48;
358             signeddectobin5(out);
359             printf("\n");
360         }
361         else{
362             out= str[i][1]-48;
363             if(str[i][2]!=0)
364                 out=out*10+str[i][2]-48;
365             dectobin5(out);
366             printf("\n");
367         }
368     }
369     if ((str[i][0]=='x' || str[i][0]=='x')&&(str[i][1]!=0)){
370         printf("1");
371         if(str[i][1]=='-'){
372             if ((str[i][2]-48>=0 &&str[i][2]-48<=9)) out=str[i]
[2]-48;
373             else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=str[i][2]-55;

```

```

374         if(str[i][3]!=0){
375             if ((str[i][3]-48>=0 &&str[i][3]-48<=9))
out=out*16+str[i][3]-48;
376             else if ((str[i][3]-65>=0 &&str[i][3]-65<=5))
out=out*16+str[i][3]-55;
377         }
378         signeddectobin5(out);
379         printf("\n");
380     }
381     else{
382         if ((str[i][1]-48>=0 &&str[i][1]-48<=9)) out=str[i]
[1]-48;
383         else if ((str[i][1]-65>=0 &&str[i][1]-65<=5))
out=str[i][1]-55;
384         if(str[i][2]!=0){
385             if ((str[i][2]-48>=0 &&str[i][2]-48<=9))
out=out*16+str[i][2]-48;
386             else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=out*16+str[i][2]-55;
387         }
388         dectobin5(out);
389         printf("\n");
390     }
391 }
392 lines++;
393 }
394 else if (strcmp(str[i],"NOT")==0){
395     opcode=3;
396     out=9;
397     dectobin4(out);
398     i++;
399     out=str[i][1]-48;
400     dectobin3(out);
401     i++;
402     out=str[i][1]-48;
403     dectobin3(out);
404     printf("11111");
405     printf("\n");
406     lines++;
407 }
408 else if (strcmp(str[i],"LD")==0){
409     opcode=4;
410     out=2;
411     dectobin4(out);
412     i++;
413     out=str[i][1]-48;
414     dectobin3(out);
415     i++;
416     if(str[i][0]=='#'){
417         if(str[i][1]=='-'){
418             out= str[i][2]-48;
419             if(str[i][3]!=0)
420                 out=out*10+str[i][3]-48;
421             if(str[i][4]!=0)
422                 out=out*10+str[i][4]-48;
423             signeddectobin9(out);
424         }
425         else{

```



```

426         out= str[i][1]-48;
427         if(str[i][2]!=0)
428             out=out*10+str[i][2]-48;
429         if(str[i][3]!=0)
430             out=out*10+str[i][3]-48;
431         dectobin9(out);
432     }
433 }
434 else{
435     for(j=0;j<1;j++){
436         if(strcmp(str[i],label[j])==0){
437             out=label2[j]-lines-1;
438             if(out>=0) {
439                 dectobin9(out);
440                 break;
441             }
442             else if(out<0){
443                 out=-out;
444                 signeddectobin9(out);
445                 break;
446             }
447         }
448     }
449 }
450 printf("\n");
451 lines++;
452 }
453 else if (strcmp(str[i],"LDR")==0){
454     opcode=5;
455     out=6;
456     dectobin4(out);
457     i++;
458     out=str[i][1]-48;
459     dectobin3(out);
460     i++;
461     out=str[i][1]-48;
462     dectobin3(out);
463     i++;
464     if(str[i][0]=='#'){
465         if(str[i][1]=='-'){
466             out= str[i][2]-48;
467             if(str[i][3]!=0)
468                 out=out*10+str[i][3]-48;
469             signeddectobin6(out);
470             printf("\n");
471         }
472         else{
473             out= str[i][1]-48;
474             if(str[i][2]!=0)
475                 out=out*10+str[i][2]-48;
476             dectobin6(out);
477             printf("\n");
478         }
479     }
480     if ((str[i][0]=='x' || str[i][0]=='x') && (str[i][1]!=0)){
481         if(str[i][1]=='-'){
482             if ((str[i][2]-48>=0 && str[i][2]-48<=9)) out=str[i]
[2]-48;

```

```

483         else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=str[i][2]-55;
484         if(str[i][3]!=0){
485             if ((str[i][3]-48>=0 &&str[i][3]-48<=9))
out=out*16+str[i][3]-48;
486             else if ((str[i][3]-65>=0 &&str[i][3]-65<=5))
out=out*16+str[i][3]-55;
487         }
488         signeddectobin6(out);
489         printf("\n");
490     }
491     else{
492         if ((str[i][1]-48>=0 &&str[i][1]-48<=9)) out=str[i]
[1]-48;
493         else if ((str[i][1]-65>=0 &&str[i][1]-65<=5))
out=str[i][1]-55;
494         if(str[i][2]!=0){
495             if ((str[i][2]-48>=0 &&str[i][2]-48<=9))
out=out*16+str[i][2]-48;
496             else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=out*16+str[i][2]-55;
497         }
498         dectobin6(out);
499         printf("\n");
500     }
501 }
502 lines++;
503 }
504 else if (strcmp(str[i],"LDI")==0){
505     opcode=4;
506     out=10;
507     dectobin4(out);
508     i++;
509     out=str[i][1]-48;
510     dectobin3(out);
511     i++;
512     if(str[i][0]=='#'){
513         if(str[i][1]=='-'){
514             out= str[i][2]-48;
515             if(str[i][3]!=0)
516                 out=out*10+str[i][3]-48;
517             if(str[i][4]!=0)
518                 out=out*10+str[i][4]-48;
519             signeddectobin9(out);
520         }
521         else{
522             out= str[i][1]-48;
523             if(str[i][2]!=0)
524                 out=out*10+str[i][2]-48;
525             if(str[i][3]!=0)
526                 out=out*10+str[i][3]-48;
527             dectobin9(out);
528         }
529     }
530     else{
531         for(j=0;j<1;j++){
532             if(strcmp(str[i],label[j])==0){
533                 out=label2[j]-lines-1;

```

```

534         if(out>=0) {
535             dectobin9(out);
536             break;
537         }
538         else if(out<0){
539             out=-out;
540             signeddectobin9(out);
541             break;
542         }
543     }
544 }
545 }
546 printf("\n");
547 lines++;
548 }
549 else if (strcmp(str[i],"LEA")==0){
550     opcode=4;
551     out=14;
552     dectobin4(out);
553     i++;
554     out=str[i][1]-48;
555     dectobin3(out);
556     i++;
557     if(str[i][0]=='#'){
558         if(str[i][1]=='-'){
559             out= str[i][2]-48;
560             if(str[i][3]!=0)
561                 out=out*10+str[i][3]-48;
562             if(str[i][4]!=0)
563                 out=out*10+str[i][4]-48;
564             signeddectobin9(out);
565         }
566         else{
567             out= str[i][1]-48;
568             if(str[i][2]!=0)
569                 out=out*10+str[i][2]-48;
570             if(str[i][3]!=0)
571                 out=out*10+str[i][3]-48;
572             dectobin9(out);
573         }
574     }
575     else{
576         for(j=0;j<1;j++){
577             if(strcmp(str[i],label[j])==0){
578                 out=label2[j]-lines-1;
579                 if(out>=0) {
580                     dectobin9(out);
581                     break;
582                 }
583                 else if(out<0){
584                     out=-out;
585                     signeddectobin9(out);
586                     break;
587                 }
588             }
589         }
590     }
591     printf("\n");

```

```

592         lines++;
593     }
594     else if (strcmp(str[i], "ST")==0){
595         opcode=4;
596         out=3;
597         dectobin4(out);
598         i++;
599         out=str[i][1]-48;
600         dectobin3(out);
601         i++;
602         if(str[i][0]=='#'){
603             if(str[i][1]=='-'){
604                 out= str[i][2]-48;
605                 if(str[i][3]!=0)
606                     out=out*10+str[i][3]-48;
607                 if(str[i][4]!=0)
608                     out=out*10+str[i][4]-48;
609                 signeddectobin9(out);
610             }
611             else{
612                 out= str[i][1]-48;
613                 if(str[i][2]!=0)
614                     out=out*10+str[i][2]-48;
615                 if(str[i][3]!=0)
616                     out=out*10+str[i][3]-48;
617                 dectobin9(out);
618             }
619         }
620     else{
621         for(j=0;j<1;j++){
622             if(strcmp(str[i],label[j])==0){
623                 out=label2[j]-lines-1;
624                 if(out>=0) {
625                     dectobin9(out);
626                     break;
627                 }
628                 else if(out<0){
629                     out=-out;
630                     signeddectobin9(out);
631                     break;
632                 }
633             }
634         }
635     }
636     printf("\n");
637     lines++;
638 }
639 else if (strcmp(str[i], "STR")==0){
640     opcode=5;
641     out=7;
642     dectobin4(out);
643     i++;
644     out=str[i][1]-48;
645     dectobin3(out);
646     i++;
647     out=str[i][1]-48;
648     dectobin3(out);
649     i++;

```

```

650         if(str[i][0]=='#'){
651             if(str[i][1]=='-'){
652                 out= str[i][2]-48;
653                 if(str[i][3]!=0)
654                     out=out*10+str[i][3]-48;
655                 signeddectobin6(out);
656                 printf("\n");
657             }
658             else{
659                 out= str[i][1]-48;
660                 if(str[i][2]!=0)
661                     out=out*10+str[i][2]-48;
662                 dectobin6(out);
663                 printf("\n");
664             }
665         }
666         if ((str[i][0]=='x' || str[i][0]=='x') && (str[i][1] != 0)) {
667             if(str[i][1]=='-'){
668                 if ((str[i][2]-48>=0 && str[i][2]-48<=9)) out=str[i]
669 [2]-48;
670                 if ((str[i][2]-65>=0 && str[i][2]-65<=5)) out=str[i]
671 [2]-55;
672                 if(str[i][2]!=0){
673                     if ((str[i][3]-48>=0 && str[i][3]-48<=9))
674 out=out*16+str[i][3]-48;
675                     if ((str[i][3]-65>=0 && str[i][3]-65<=5))
676 out=out*16+str[i][3]-55;
677                 }
678                 signeddectobin6(out);
679                 printf("\n");
680             }
681             else{
682                 if ((str[i][1]-48>=0 && str[i][1]-48<=9)) out=str[i]
683 [1]-48;
684                 if ((str[i][1]-65>=0 && str[i][1]-65<=5)) out=str[i]
685 [1]-55;
686                 if(str[i][2]!=0){
687                     if ((str[i][2]-48>=0 && str[i][2]-48<=9))
688 out=out*16+str[i][2]-48;
689                     if ((str[i][2]-65>=0 && str[i][2]-65<=5))
690 out=out*16+str[i][2]-55;
691                 }
692                 dectobin6(out);
693                 printf("\n");
694             }
695         }
696         lines++;
697     }
698     else if (strcmp(str[i], "STI")==0){
699         opcode=4;
700         out=11;
701         dectobin4(out);
702         i++;
703         out=str[i][1]-48;
704         dectobin3(out);
705         i++;
706         if(str[i][0]=='#'){
707             if(str[i][1]=='-'){

```

```

700         out= str[i][2]-48;
701         if(str[i][3]!=0)
702             out=out*10+str[i][3]-48;
703         if(str[i][4]!=0)
704             out=out*10+str[i][4]-48;
705         signeddectobin9(out);
706     }
707     else{
708         out= str[i][1]-48;
709         if(str[i][2]!=0)
710             out=out*10+str[i][2]-48;
711         if(str[i][3]!=0)
712             out=out*10+str[i][3]-48;
713         dectobin9(out);
714     }
715 }
716 else{
717     for(j=0;j<1;j++){
718         if(strcmp(str[i],label[j])==0){
719             out=label2[j]-lines-1;
720             if(out>=0) {
721                 dectobin9(out);
722                 break;
723             }
724             else if(out<0){
725                 out=-out;
726                 signeddectobin9(out);
727                 break;
728             }
729         }
730     }
731 }
732 printf("\n");
733 lines++;
734 }
735 else if (strcmp(str[i],"TRAP")==0){
736     opcode=11;
737     printf("11110000");
738     i++;
739     if ((str[i][1]-48>=0 &&str[i][1]-48<=9)) out=str[i][1]-48;
740     else if ((str[i][1]-65>=0 &&str[i][1]-65<=5)) out=str[i]
[1]-55;
741     if(str[i][2]!=0){
742         if ((str[i][2]-48>=0 &&str[i][2]-48<=9))
out=out*16+str[i][2]-48;
743         if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=out*16+str[i][2]-55;
744     }
745     dectobin8(out);
746     printf("\n");
747     lines++;
748 }
749 else if (strcmp(str[i],"HALT")==0){
750     opcode=12;
751     printf("1111000000100101");
752     printf("\n");
753     lines++;
754 }

```

```

755     else if (strcmp(str[i], "JMP")==0){
756         opcode=13;
757         out=12;
758         dectobin4(out);
759         printf("000");
760         i++;
761         out=str[i][1]-48;
762         dectobin3(out);
763         printf("000000");
764         printf("\n");
765         lines++;
766     }
767     else if (strcmp(str[i], "RET")==0){
768         opcode=14;
769         printf("1100000111000000");
770         printf("\n");
771         lines++;
772     }
773     else if (strcmp(str[i], "JSR")==0){
774         opcode=15;
775         out=4;
776         dectobin4(out);
777         i++;
778         if(str[i][0]=='#'){
779             printf("1");
780             if(str[i][1]=='-'){
781                 out= str[i][2]-48;
782                 if(str[i][3]!=0)
783                     out=out*10+str[i][3]-48;
784                 if(str[i][4]!=0)
785                     out=out*10+str[i][4]-48;
786                 if(str[i][5]!=0)
787                     out=out*10+str[i][5]-48;
788                 signeddectobin11(out);
789             }
790             else{
791                 out= str[i][1]-48;
792                 if(str[i][2]!=0)
793                     out=out*10+str[i][2]-48;
794                 if(str[i][3]!=0)
795                     out=out*10+str[i][3]-48;
796                 if(str[i][4]!=0)
797                     out=out*10+str[i][4]-48;
798                 dectobin9(out);
799             }
800         }
801         else{
802             printf("1");
803             for(j=0; j<1; j++){
804                 if(strcmp(str[i], label[j])==0){
805                     out=label2[j]-lines-1;
806                     if(out>=0) {
807                         dectobin11(out);
808                         break;
809                     }
810                     else if(out<0){
811                         out=-out;
812                         signeddectobin11(out);

```

```

813         break;
814     }
815 }
816 }
817 }
818 // i++;
819 lines++;
820 printf("\n");
821 }
822 else if (strcmp(str[i], "JSRR")==0){
823     opcode=16;
824     printf("0100000") ;
825     i++;
826     out=str[i][1]-48;
827     dectobin3(out);
828     printf("000000");
829     printf("\n");
830     lines++;
831 }
832 else if (strcmp(str[i], "RTI")==0){
833     opcode=17;
834     printf("1000000000000000");
835     printf("\n");
836     lines++;
837 }
838 else if (strcmp(str[i], ".FILL")==0){
839     opcode=18;
840     i++;
841     if(str[i][0]=='#'){
842         if(str[i][1]=='-'){
843             out= str[i][2]-48;
844             if(str[i][3]!=0)
845                 out=out*10+str[i][3]-48;
846             if(str[i][4]!=0)
847                 out=out*10+str[i][4]-48;
848             if(str[i][5]!=0)
849                 out=out*10+str[i][5]-48;
850             if(str[i][6]!=0)
851                 out=out*10+str[i][6]-48;
852             signeddectobin16(out);
853             printf("\n");
854         }
855         else{
856             out= str[i][1]-48;
857             if(str[i][2]!=0)
858                 out=out*10+str[i][2]-48;
859             if(str[i][3]!=0)
860                 out=out*10+str[i][3]-48;
861             if(str[i][4]!=0)
862                 out=out*10+str[i][4]-48;
863             if(str[i][5]!=0)
864                 out=out*10+str[i][5]-48;
865             dectobin16(out);
866             printf("\n");
867         }
868     }
869     if ((str[i][0]=='x' || str[i][0]=='X') && (str[i][1]!=0)){
870         if(str[i][1]=='-'){

```



```

871         if ((str[i][2]-48>=0 &&str[i][2]-48<=9)) out=str[i]
[2]-48;
872         else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=str[i][2]-55;
873         if(str[i][3]!=0){
874             if ((str[i][3]-48>=0 &&str[i][3]-48<=9))
out=out*16+str[i][3]-48;
875             else if ((str[i][3]-65>=0 &&str[i][3]-65<=5))
out=out*16+str[i][3]-55;
876             if(str[i][4]!=0){
877                 if ((str[i][4]-48>=0 &&str[i][4]-48<=9))
out=out*16+str[i][4]-48;
878                 else if ((str[i][4]-65>=0 &&str[i][4]-65<=5))
out=out*16+str[i][4]-55;
879                 if(str[i][5]!=0){
880                     if ((str[i][5]-48>=0 &&str[i][5]-48<=9))
out=out*16+str[i][5]-48;
881                     else if ((str[i][5]-65>=0 &&str[i]
[5]-65<=5)) out=out*16+str[i][5]-55;
882                 }
883             }
884         }
885         signeddectobin16(out);
886         printf("\n");
887     }
888     else{
889         if ((str[i][1]-48>=0 &&str[i][1]-48<=9)) out=str[i]
[1]-48;
890         else if ((str[i][1]-65>=0 &&str[i][1]-65<=5))
out=str[i][1]-55;
891         if(str[i][2]!=0){
892             if ((str[i][2]-48>=0 &&str[i][2]-48<=9))
out=out*16+str[i][2]-48;
893             else if ((str[i][2]-65>=0 &&str[i][2]-65<=5))
out=out*16+str[i][2]-55;
894             if(str[i][3]!=0){
895                 if ((str[i][3]-48>=0 &&str[i][3]-48<=9))
out=out*16+str[i][3]-48;
896                 else if ((str[i][3]-65>=0 &&str[i][3]-65<=5))
out=out*16+str[i][3]-55;
897                 if(str[i][4]!=0){
898                     if ((str[i][4]-48>=0 &&str[i][4]-48<=9))
out=out*16+str[i][4]-48;
899                     else if ((str[i][4]-65>=0 &&str[i]
[4]-65<=5)) out=out*16+str[i][4]-55;
900                 }
901             }
902         }
903         dectobin16(out);
904         printf("\n");
905     }
906 }
907 lines++;
908 }
909 else if (strcmp(str[i],".BLKW")==0){
910     opcode=19;
911     i++;
912     j=1;

```

```

913         out=0;
914         while(str[i][j]!=0){
915             out=out*10+str[i][j]-48;
916             j++;
917         }
918
919         for(j=0;j<out;j++){
920             printf("0111011101110111");
921             printf("\n");
922         }
923         lines+=out;
924     }
925     else if (strcmp(str[i],".STRINGZ")==0){
926         opcode=20;
927         p=0;
928         i++;
929         j=1;
930         while(str[i][j]!=''){
931             if(str[i][j]!=0){
932                 out=str[i][j];
933                 dectobin16(out);
934                 j++;
935                 printf("\n");
936                 p++;
937             }
938             else {
939                 printf("0000000000100000");
940                 printf("\n");
941                 i++;
942                 p++;
943                 j=0;
944             }
945         }
946         printf("0000000000000000");
947         printf("\n");
948         lines+=p+1;
949     }
950     else if (strcmp(str[i],"GETC")==0){
951         opcode=12;
952         printf("1111000000100000");
953         printf("\n");
954         lines++;
955     }
956     else if (strcmp(str[i],"OUT")==0){
957         opcode=12;
958         printf("1111000000100001");
959         printf("\n");
960         lines++;
961     }
962     else if (strcmp(str[i],"PUTS")==0){
963         opcode=12;
964         printf("1111000000100010");
965         printf("\n");
966         lines++;
967     }
968     else if (strcmp(str[i],"IN")==0){
969         opcode=12;
970         printf("1111000000100011");

```

```

971         printf("\n");
972         lines++;
973     }
974     else if (strcmp(str[i], "PUTSP")==0){
975         opcode=12;
976         printf("1111000000100100");
977         printf("\n");
978         lines++;
979     }
980     else if
981 (strcmp(str[i], "BRn")==0 || strcmp(str[i], "BRnz")==0 || strcmp(str[i], "BRnp")==
982 =0 || strcmp(str[i], "BRnzp")==0 || strcmp(str[i], "BRz")==0 || strcmp(str[i], "BRz
983 p")==0 || strcmp(str[i], "BRp")==0 || strcmp(str[i], "BR")==0){
984         printf("0000");
985         if(str[i][2]==0) printf("111");
986         else if(str[i][2]=='p') printf("001");
987         else if(str[i][2]=='z'){
988             printf("01");
989             if(str[i][3]=='p') printf("1");
990             else printf("0");
991         }
992         else if(str[i][2]=='n'){
993             printf("1");
994             if(str[i][3]=='p') printf("01");
995             else if(str[i][3]=='z'){
996                 printf("1");
997                 if(str[i][4]=='p') printf("1");
998                 else printf("0");
999             }
1000             else printf("00");
1001         }
1002         i++;
1003         if(str[i][0]=='#'){
1004             if(str[i][1]=='-'){
1005                 out= str[i][2]-48;
1006                 if(str[i][3]!=0)
1007                     out=out*10+str[i][3]-48;
1008                 if(str[i][4]!=0)
1009                     out=out*10+str[i][4]-48;
1010                 signeddectobin9(out);
1011             }
1012             else{
1013                 out= str[i][1]-48;
1014                 if(str[i][2]!=0)
1015                     out=out*10+str[i][2]-48;
1016                 if(str[i][3]!=0)
1017                     out=out*10+str[i][3]-48;
1018                 dectobin9(out);
1019             }
1020         }
1021         else{
1022             for(j=0;j<1;j++){
1023                 if(strcmp(str[i], label[j])==0){
1024                     out=label[j]-lines-1;
1025                     if(out>=0) {
1026                         dectobin9(out);
1027                         break;
1028                     }
1029                 }
1030             }
1031         }
1032     }
1033 }
1034 }
1035 }
1036 }
1037 }
1038 }
1039 }
1040 }
1041 }
1042 }
1043 }
1044 }
1045 }
1046 }
1047 }
1048 }
1049 }
1050 }
1051 }
1052 }
1053 }
1054 }
1055 }
1056 }
1057 }
1058 }
1059 }
1060 }
1061 }
1062 }
1063 }
1064 }
1065 }
1066 }
1067 }
1068 }
1069 }
1070 }
1071 }
1072 }
1073 }
1074 }
1075 }
1076 }
1077 }
1078 }
1079 }
1080 }
1081 }
1082 }
1083 }
1084 }
1085 }
1086 }
1087 }
1088 }
1089 }
1090 }
1091 }
1092 }
1093 }
1094 }
1095 }
1096 }
1097 }
1098 }
1099 }
1100 }
1101 }
1102 }
1103 }
1104 }
1105 }
1106 }
1107 }
1108 }
1109 }
1110 }
1111 }
1112 }
1113 }
1114 }
1115 }
1116 }
1117 }
1118 }
1119 }
1120 }
1121 }
1122 }
1123 }
1124 }
1125 }
1126 }
1127 }
1128 }
1129 }
1130 }
1131 }
1132 }
1133 }
1134 }
1135 }
1136 }
1137 }
1138 }
1139 }
1140 }
1141 }
1142 }
1143 }
1144 }
1145 }
1146 }
1147 }
1148 }
1149 }
1150 }
1151 }
1152 }
1153 }
1154 }
1155 }
1156 }
1157 }
1158 }
1159 }
1160 }
1161 }
1162 }
1163 }
1164 }
1165 }
1166 }
1167 }
1168 }
1169 }
1170 }
1171 }
1172 }
1173 }
1174 }
1175 }
1176 }
1177 }
1178 }
1179 }
1180 }
1181 }
1182 }
1183 }
1184 }
1185 }
1186 }
1187 }
1188 }
1189 }
1190 }
1191 }
1192 }
1193 }
1194 }
1195 }
1196 }
1197 }
1198 }
1199 }
1200 }
1201 }
1202 }
1203 }
1204 }
1205 }
1206 }
1207 }
1208 }
1209 }
1210 }
1211 }
1212 }
1213 }
1214 }
1215 }
1216 }
1217 }
1218 }
1219 }
1220 }
1221 }
1222 }
1223 }
1224 }
1225 }
1226 }
1227 }
1228 }
1229 }
1230 }
1231 }
1232 }
1233 }
1234 }
1235 }
1236 }
1237 }
1238 }
1239 }
1240 }
1241 }
1242 }
1243 }
1244 }
1245 }
1246 }
1247 }
1248 }
1249 }
1250 }
1251 }
1252 }
1253 }
1254 }
1255 }
1256 }
1257 }
1258 }
1259 }
1260 }
1261 }
1262 }
1263 }
1264 }
1265 }
1266 }
1267 }
1268 }
1269 }
1270 }
1271 }
1272 }
1273 }
1274 }
1275 }
1276 }
1277 }
1278 }
1279 }
1280 }
1281 }
1282 }
1283 }
1284 }
1285 }
1286 }
1287 }
1288 }
1289 }
1290 }
1291 }
1292 }
1293 }
1294 }
1295 }
1296 }
1297 }
1298 }
1299 }
1300 }
1301 }
1302 }
1303 }
1304 }
1305 }
1306 }
1307 }
1308 }
1309 }
1310 }
1311 }
1312 }
1313 }
1314 }
1315 }
1316 }
1317 }
1318 }
1319 }
1320 }
1321 }
1322 }
1323 }
1324 }
1325 }
1326 }
1327 }
1328 }
1329 }
1330 }
1331 }
1332 }
1333 }
1334 }
1335 }
1336 }
1337 }
1338 }
1339 }
1340 }
1341 }
1342 }
1343 }
1344 }
1345 }
1346 }
1347 }
1348 }
1349 }
1350 }
1351 }
1352 }
1353 }
1354 }
1355 }
1356 }
1357 }
1358 }
1359 }
1360 }
1361 }
1362 }
1363 }
1364 }
1365 }
1366 }
1367 }
1368 }
1369 }
1370 }
1371 }
1372 }
1373 }
1374 }
1375 }
1376 }
1377 }
1378 }
1379 }
1380 }
1381 }
1382 }
1383 }
1384 }
1385 }
1386 }
1387 }
1388 }
1389 }
1390 }
1391 }
1392 }
1393 }
1394 }
1395 }
1396 }
1397 }
1398 }
1399 }
1400 }
1401 }
1402 }
1403 }
1404 }
1405 }
1406 }
1407 }
1408 }
1409 }
1410 }
1411 }
1412 }
1413 }
1414 }
1415 }
1416 }
1417 }
1418 }
1419 }
1420 }
1421 }
1422 }
1423 }
1424 }
1425 }
1426 }
1427 }
1428 }
1429 }
1430 }
1431 }
1432 }
1433 }
1434 }
1435 }
1436 }
1437 }
1438 }
1439 }
1440 }
1441 }
1442 }
1443 }
1444 }
1445 }
1446 }
1447 }
1448 }
1449 }
1450 }
1451 }
1452 }
1453 }
1454 }
1455 }
1456 }
1457 }
1458 }
1459 }
1460 }
1461 }
1462 }
1463 }
1464 }
1465 }
1466 }
1467 }
1468 }
1469 }
1470 }
1471 }
1472 }
1473 }
1474 }
1475 }
1476 }
1477 }
1478 }
1479 }
1480 }
1481 }
1482 }
1483 }
1484 }
1485 }
1486 }
1487 }
1488 }
1489 }
1490 }
1491 }
1492 }
1493 }
1494 }
1495 }
1496 }
1497 }
1498 }
1499 }
1500 }
1501 }
1502 }
1503 }
1504 }
1505 }
1506 }
1507 }
1508 }
1509 }
1510 }
1511 }
1512 }
1513 }
1514 }
1515 }
1516 }
1517 }
1518 }
1519 }
1520 }
1521 }
1522 }
1523 }
1524 }
1525 }
1526 }
1527 }
1528 }
1529 }
1530 }
1531 }
1532 }
1533 }
1534 }
1535 }
1536 }
1537 }
1538 }
1539 }
1540 }
1541 }
1542 }
1543 }
1544 }
1545 }
1546 }
1547 }
1548 }
1549 }
1550 }
1551 }
1552 }
1553 }
1554 }
1555 }
1556 }
1557 }
1558 }
1559 }
1560 }
1561 }
1562 }
1563 }
1564 }
1565 }
1566 }
1567 }
1568 }
1569 }
1570 }
1571 }
1572 }
1573 }
1574 }
1575 }
1576 }
1577 }
1578 }
1579 }
1580 }
```

```

1026         else if(out<0){
1027             out=-out;
1028             signeddectobin9(out);
1029             break;
1030         }
1031     }
1032 }
1033 }
1034 printf("\n");
1035 lines++;
1036 }
1037 }
1038
1039 // for(i=0;i<1;i++){
1040 //     printf("%d ",label2[i]);
1041 //     printf("%s",label[i]);
1042 //     printf("\n");
1043 // }
1044
1045 return 0;
1046 }
1047
1048 //将十进制数转换为十六位二进制数输出
1049 void dectobin16(int n){
1050     int yu=0;
1051     int i=0;
1052     int a[15]={};
1053     for(i=0;i<=15;i++) {
1054         yu = n%2;
1055         a[15-i]=yu;
1056         n/=2;
1057     }
1058     for(i=0;i<=15;i++){
1059         printf("%d",a[i]);
1060     }
1061 }
1062
1063 //将十进制数转换为四位二进制数输出
1064 void dectobin4(int n){
1065     int yu=0;
1066     int i=0;
1067     int a[3]={};
1068     for(i=0;i<=3;i++) {
1069         yu = n%2;
1070         a[3-i]=yu;
1071         n/=2;
1072     }
1073     for(i=0;i<=3;i++){
1074         printf("%d",a[i]);
1075     }
1076 }
1077
1078 //将十进制数转换为三位二进制数输出
1079 void dectobin3(int n){
1080     int yu=0;
1081     int i=0;
1082     int a[3]={};
1083     for(i=0;i<=2;i++) {

```

```

1084     yu = n%2;
1085     a[2-i]=yu;
1086 //     printf("%d %d\n",a[2-i],2-i);
1087     n/=2;
1088 }
1089 for(i=0;i<=2;i++){
1090     printf("%d",a[i]);
1091 }
1092 // printf("\n");
1093 }
1094
1095 //将十进制数转换为五位二进制数输出
1096 void dectobin5(int n){
1097     int yu=0;
1098     int i=0;
1099     int a[4]={};
1100     for(i=0;i<=4;i++) {
1101         yu = n%2;
1102         a[4-i]=yu;
1103         n/=2;
1104     }
1105     for(i=0;i<=4;i++){
1106         printf("%d",a[i]);
1107     }
1108 }
1109
1110 //将十进制数转换为九位二进制数输出
1111 void dectobin9(int n){
1112     int yu=0;
1113     int i=0;
1114     int a[8]={};
1115     for(i=0;i<=8;i++) {
1116         yu = n%2;
1117         a[8-i]=yu;
1118         n/=2;
1119     }
1120     for(i=0;i<=8;i++){
1121         printf("%d",a[i]);
1122     }
1123 }
1124
1125 //将十进制数转换为六位二进制数输出
1126 void dectobin6(int n){
1127     int yu=0;
1128     int i=0;
1129     int a[5]={};
1130     for(i=0;i<=5;i++) {
1131         yu = n%2;
1132         a[5-i]=yu;
1133         n/=2;
1134     }
1135     for(i=0;i<=5;i++){
1136         printf("%d",a[i]);
1137     }
1138 }
1139
1140 //将十进制数转换为八位二进制数输出
1141 void dectobin8(int n){

```

```

1142     int yu=0;
1143     int i=0;
1144     int a[7]={};
1145     for(i=0;i<=7;i++) {
1146         yu = n%2;
1147         a[7-i]=yu;
1148         n/=2;
1149     }
1150     for(i=0;i<=7;i++){
1151         printf("%d",a[i]);
1152     }
1153 }
1154
1155 //将十进制数转换为十一位二进制数输出
1156 void dectobin11(int n){
1157     int yu=0;
1158     int i=0;
1159     int a[10]={};
1160     for(i=0;i<=10;i++) {
1161         yu = n%2;
1162         a[10-i]=yu;
1163         n/=2;
1164     }
1165     for(i=0;i<=10;i++){
1166         printf("%d",a[i]);
1167     }
1168 }
1169
1170 //将十进制负数转换为五位二进制数输出
1171 void signeddectobin5(int n){
1172     int yu=0;
1173     int i=0;
1174     int a[4]={};
1175     for(i=0;i<=4;i++) {
1176         yu = n%2;
1177         a[4-i]=yu;
1178         n/=2;
1179     }
1180     for(i=0;i<=4;i++) {
1181         if (a[i]==0) a[i]=1;
1182         else if (a[i]==1) a[i]=0;
1183     }
1184     i=4;
1185     if(a[i]==0) a[i]=1;
1186     else{
1187         while(a[i]==1){
1188             a[i]=0;
1189             i--;
1190         }
1191         if(i>=0) a[i]=1;
1192     }
1193     for(i=0;i<=4;i++){
1194         printf("%d",a[i]);
1195     }
1196 }
1197
1198 //将十进制负数转换为六位二进制数输出
1199 void signeddectobin6(int n){

```

```

1200     int yu=0;
1201     int i=0;
1202     int a[5]={};
1203     for(i=0;i<=5;i++) {
1204         yu = n%2;
1205         a[5-i]=yu;
1206         n/=2;
1207     }
1208     for(i=0;i<=5;i++) {
1209         if (a[i]==0) a[i]=1;
1210         else if (a[i]==1) a[i]=0;
1211     }
1212     i=5;
1213     if(a[i]==0) a[i]=1;
1214     else{
1215         while(a[i]==1){
1216             a[i]=0;
1217             i--;
1218         }
1219         if(i>=0) a[i]=1;
1220     }
1221     for(i=0;i<=5;i++){
1222         printf("%d",a[i]);
1223     }
1224 }
1225
1226 //将十进制负数转换为十六位二进制数输出
1227 void signeddectobin16(int n){
1228     int yu=0;
1229     int i=0;
1230     int a[15]={};
1231     for(i=0;i<=15;i++) {
1232         yu = n%2;
1233         a[15-i]=yu;
1234         n/=2;
1235     }
1236     for(i=0;i<=15;i++) {
1237         if (a[i]==0) a[i]=1;
1238         else if (a[i]==1) a[i]=0;
1239     }
1240     i=15;
1241     if(a[i]==0) a[i]=1;
1242     else{
1243         while(a[i]==1){
1244             a[i]=0;
1245             i--;
1246         }
1247         if(i>=0) a[i]=1;
1248     }
1249     for(i=0;i<=15;i++){
1250         printf("%d",a[i]);
1251     }
1252 }
1253
1254 //将十进制负数转换为九位二进制数输出
1255 void signeddectobin9(int n){
1256     int yu=0;
1257     int i=0;

```

```

1258     int a[8]={};
1259     for(i=0;i<=8;i++) {
1260         yu = n%2;
1261         a[8-i]=yu;
1262         n/=2;
1263     }
1264     for(i=0;i<=8;i++) {
1265         if (a[i]==0) a[i]=1;
1266         else if (a[i]==1) a[i]=0;
1267     }
1268     i=8;
1269     if(a[i]==0) a[i]=1;
1270     else{
1271         while(a[i]==1){
1272             a[i]=0;
1273             i--;
1274         }
1275         if(i>=0) a[i]=1;
1276     }
1277     for(i=0;i<=8;i++){
1278         printf("%d",a[i]);
1279     }
1280 }
1281
1282 //将十进制负数转换为十一位二进制数输出
1283 void signedtobin11(int n){
1284     int yu=0;
1285     int i=0;
1286     int a[10]={};
1287     for(i=0;i<=10;i++) {
1288         yu = n%2;
1289         a[10-i]=yu;
1290         n/=2;
1291     }
1292     for(i=0;i<=10;i++) {
1293         if (a[i]==0) a[i]=1;
1294         else if (a[i]==1) a[i]=0;
1295     }
1296     i=10;
1297     if(a[i]==0) a[i]=1;
1298     else{
1299         while(a[i]==1){
1300             a[i]=0;
1301             i--;
1302         }
1303         if(i>=0) a[i]=1;
1304     }
1305     for(i=0;i<=10;i++){
1306         printf("%d",a[i]);
1307     }
1308 }
1309
1310

```


Q&A

null