26. Write a yacc program to check the acceptance of the language a^n.b^n where n>=0.

```
//lex
%{
#include "y.tab.h"
%}

%%
[aA] {return A;}
[bB] {return B;}
\n {return NL;}
. {return yytext[0];}
%%

//yacc
%{
#include<stdio.h>
#include<stdlib.h>
%}

%token A B NL

%%
stmt: S NL {printf("valid string\n");
            exit(0);}
;
S: A S B |
;
%%

int yyerror(char *msg)
{
printf("invalid string\n");
exit(0);
}
main()
{
printf("enter the string\n");
yyparse();
}
```

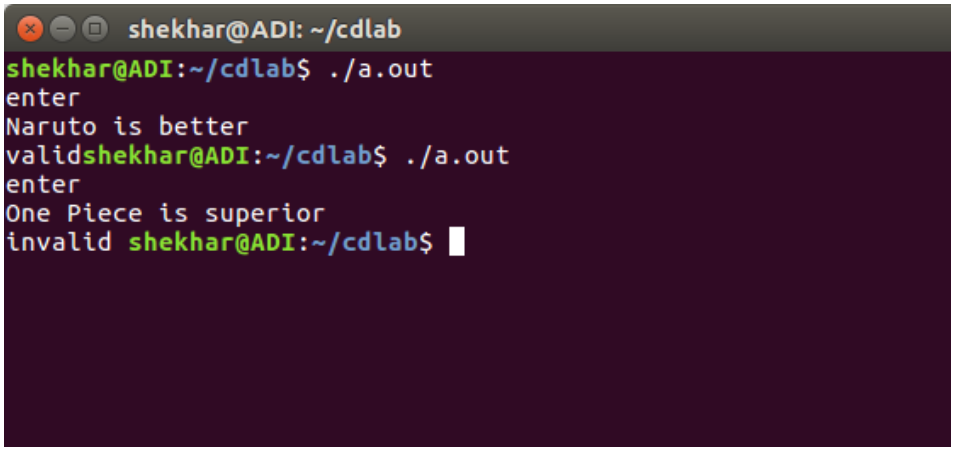
OUTPUT:

27. Implement a sentence parser using yacc for recognizing English sentences.

```
//lex
%{
#include "y.tab.h"
%}
%%
Naruto|DBZ|DeathNote {return E;}
is {return V;}
better|superior {return O; }
[ \t] {;}
\n {return 0;}
. {return yytext[0];}
%%

//yacc
%{
#include<stdio.h>
%}
%token E V O
%%
stmt: S
    ;
S: E S V O
  |
 ;
%%
void main()
{
printf("enter \n");
yyparse();
printf("valid");
exit(0);
}
void yyerror()
{
printf("invalid ");
exit(0);
}
```

OUTPUT:



28. Write a program in yacc to implement binary to decimal conversion.

```
//lex
%{
#include<stdio.h>
#include<stdlib.h>
#include"y.tab.h"
extern int yylval;
%}
%%
0 {yylval=0;return ZERO;}
1 {yylval=1;return ONE;}

[ \t] {;}
\n return 0;
. return yytext[0];
%%

//yacc
%{
#include<stdio.h>
#include<stdlib.h>
void yyerror(char *s);
%}
%token ZERO ONE
%%
N: L {printf("\n%d",$$);}
L: L B {$$=$1*2+$2;}
| B {$$=$1;}
B:ZERO {$$=$1;}
|ONE {$$=$1;};
%%
int main()
{
while(yyparse());
}
yyerror(char *s)
{
fprintf(stdout,"\n%s",s);
}
```

OUTPUT: