```
#include "bits/stdc++.h"

using namespace std;


vector<pair<string, int>> opcode = {{"ADD",0},{"SUB",1},{"ADDi",2},{"AND",3},{"OR",4},{"Sll",5},{"Slr",6},{"LW",7},{"SW",8},{"IN",9},{"OUT",10}

,{"Beq",11},{"Slt",12},{"Slti",13},{"J",14},{"Not",15}};


map<int,string> binn = {{0,"0000"},{1,"0001"},{2,"0010"},{3,"0011"},{4,"0100"},{5,"0101"},{6,"0110"},{7,"0111"},{8,"1000"},{9,"1001"},{10,"1010"}

,{11,"1011"},{12,"1100"},{13,"1101"},{14,"1110"},{15,"1111"}};


map<string,string> hexx = {{"0000","0"},{"0001","1"},{"0010","2"},{"0011","3"},{"0100","4"},{"0101","5"},{"0110","6"},{"0111","7"},{"1000","8"},{"1001","9"},{"1010","A"}

,{"1011","B"},{"1100","C"},{"1101","D"},{"1110","E"},{"1111","F"}};


string decToBin(int n)

{

// array to store binary number

int binaryNum[32];

string s;

// counter for binary array

int i = 0;

while (n > 0) {


// storing remainder in binary array

binaryNum[i] = n % 2;

n = n / 2;

i++;
```

```
}

// printing binary array in reverse order
for(int j=8-i-1;j>=0;j--)s+="0";
for (int j = i - 1; j >= 0; j--)
s+=to_string(binaryNum[j]);
return s;
}

vector<pair<string, string>> reg = {{"$zero",
"0000"},{"$t0","0001"},{"$t1","0010"},{"$t2","0011"},{"$t3","0100"},{"$t4","0101"},{"$s0","0110"},{"$s1
","0111"},{"$s2","1000"},{"$s3","1001"},{"$s4","1010"}
,{"$s5","1011"},{"$s6","1100"},{"","1101"},{"","1110"},{"","1111"}};

map<int,int> lim = {{0,2},{1,2},{2,1},{3,2},{4,2},{5,1},{6,1},{7,1},{8,1},{9,1},{10,1},{11,1},{12,2},{13,1},{14,0},{15,2}};
map<int,int> lima = {{0,3},{1,3},{2,3},{3,3},{4,3},{5,3},{6,3},{7,3},{8,3},{9,3},{10,3},{11,3},{12,3},{13,3},{14,2},{15,3}};

string pars(string s){
string ss="";
string ss2="";

int i=0;
for(i=0;i<s.size();i++){
if(s[i]!='(')ss+=s[i];
else break;
}
```

```
for(i=i+1;i<s.size();i++){
if(s[i]!=')')ss2+=s[i];
else break;
}
string fin="";
//cout<<fin<<endl;
int flag2=0;
for(auto i:reg){
if(ss2.compare(i.first)==0){flag2=1;fin+=i.second;}
}
if(flag2==1 and lim[stoi(ss)]<16)fin+=binn[stoi(ss)];
else fin="-1";
return fin;
}

int main(){
string s;

string final="";
int flag=0;
int cnt=0;
//freopen("input.txt","r",stdin);
//freopen("output.txt","w",stdout);
while(getline(cin,s)){
flag = -1;
stringstream br(s);
string word, opc;

int rn=0, arg=0;
```

```
if(cnt)
final+="\n";
cnt++;
while(br>>word){

arg++;

if(flag==-1){
for(auto i:opcode){
if(word.compare(i.first)==0){flag=i.second,opc = i.second ,arg++,final+=hexx[binn[i.second]]+" ";}
}
if(flag!=-1)continue;
}
if(flag==-1){cout<<"error 1 invalid opcode"<<endl;
return 0;}

if(flag >=0 ){
// cout<<flag<<endl;
if(flag==7 || flag==8){
string fin=pars(word);
if(fin=="-1"){cout<<"error 3 register limit for operation exceeded"<<endl;return 0;}
else {

final+=hexx[fin.substr(0,4)]+" ";
final+=hexx[fin.substr(4,8)];
}
arg++;

}
```

```
else{
int flag2=0;
for(auto i:reg){

if(word.compare(i.first)==0){flag2=1,rn++,final+=hexx[i.second]+" ";}
}
if(rn>lim[flag]){cout<<"error 3 register limit for operation exceeded"<<endl;return 0;}

if(flag2==1) continue;
//          cout<<flag<<endl;
if((flag==2 || flag == 13 || flag>=5 and flag <=11) and rn==1)  {
//cout<<word<<" "<<rn<<endl;
int xx = stoi(word);
if(xx>=16){cout<<"error 4 integet limit is up to 15"<<endl;return 0;}
final+=hexx[binn[xx]];
flag2=1;
}
else if(flag==14 and rn==0){
int xx = stoi(word);
if(xx>=16){cout<<"error 4 integet limit is up to 15"<<endl;return 0;}
if(hexx[decToBin(xx).substr(0,4)]!="0")
final+=hexx[decToBin(xx).substr(0,4)];
final+=hexx[decToBin(xx).substr(4,8)];
flag2=1;
}
if(flag2==0){cout<<"error 2 invalid token"<<endl;return 0;}

}
}
```

```
}
//cout<<flag<<" "<<arg<<endl;
if(lima[flag]!=arg-1){cout<<"error 5 invalid no of arguments"<<endl;return 0;}


}


cout<<final<<endl;
return 0;
}
```