Diffchecker

- 145 Removals + 102 Additions

12.cpp

```
#include <iostream>
#include <iomanip>
#include <string>
#include <sstream>

using namespace std;

const int maxCandidate=10;//# of candidate max
int candidate=0;//# of candidate we count
string nameList[maxCandidate];//candidate name list
int voteList[maxCandidate]={0,0,0,0,0,0,0,0,0,0};//vote list
int tmpVoteList[maxCandidate]={0,0,0,0,0,0,0,0,0,0};//temp vote list b4 check its valid or not
int total=0;//total vote
int valid=0;//valid vote
int spoilt=0;//spoilt vote
int voteAllow=0;//vote allowed each person
int countUp=0;//temp vote count up
int argFlag=0;//check if theres a argument or not
int voteNow=0;//a boolean tells we done with names lets begin with count votes

/*********** validVote *************
     Purpose: This function will be called after read a valid vote
     Returns: This function returns nothing
***********************************/
void validVote()
{
  for(int j=0; j<candidate; j++)
  {
    voteList[j]+=tmpVoteList[j];
    //cout << tmpVoteList[j];
    tmpVoteList[j]=0;
  }
  //cout << endl;
  countUp=0;
  valid++;
  total++;
}
/*********** spoiltVote *************
     Purpose: This function will be called after read a spoilt vote
     Returns: This function returns nothing
************************************/
void spoiltVote()
{
  for(int j=0; j<10; j++)
    tmpVoteList[j]=0;
  countUp=0;
  spoilt++;
```

```
#include <iostream>
#include <iomanip>
#include <string>
#include <sstream>

using namespace std;

const int maxChoice = 10;//# of numchoice max
int numchoice = 0;//# of numchoice we count
string names [maxChoice];//numchoice name list
int numVotes [maxChoice] = {0,0,0,0,0,0,0,0,0,0};//vote list
int tempVotes [maxChoice] = {0,0,0,0,0,0,0,0,0,0};//temp vote list b4 check its validVotes or not
int totalVotes = 0;//totalVotes   vote
int validVotes = 0;//validVotes vote
int spoiledVote = 0;//spoiledVote vote
int allowedVotes = 0;//vote allowed each person
int incCount = 0;//temp vote count up
int isFlag = 0;//check if theres a argument or not
int startVote = 0;//a boolean tells we done with names lets begin with count votes

// This fucntion reads valid votes and counts them
void readValidVotes() {
  for(int i = 0; i < numchoice; i++) {
    numVotes[i]+=tempVotes[i];
    tempVotes[i]=0;
  }
  incCount=0;
  validVotes++;
  totalVotes + +;
}
//This function reads votes and counts spoiled votes
void spoiledVoteCount() {
  for(int i=0; i<10; i++) {
    tempVotes[i]=0;
  }
  incCount=0;
  spoiledVote++;
  totalVotes++;
  total++;
}
/*********** checkVote *************
     Purpose: This function checks if the vote is valid or not
     Returns: This function returns nothing but will change global variables which counts votes
***********************************/
void checkVote(string ticket)
{
   int voteVoter=0;
   int countUp=0;
   stringstream maStringStream(ticket);
   int n;
   while(maStringStream >> n)
   {
     voteVoter++;
     tmpVoteList[voteVoter-1]+=n;
     countUp+=n;
   }
  if((countUp>voteAllow) || (voteVoter != candidate))
    spoiltVote();
  else
    validVote();
}
/*********** addCandidate **********
     Purpose: This function consume a string which is candidate's name and we store it in the name list
     Returns: This function returns nothing but will change global variable which is the name list
***********************************/
void addCandidate(string name)
{
  if(candidate<10)
  {
    nameList[candidate]=name;
    candidate++;
    if(argFlag==0)
      voteAllow++;
  }
}
/*********** nameOrVote ************
     Purpose: This function consume a string and check if its a candidate name or vote
     Returns: This function returns nothing but will call vote/name function
***********************************/
void nameOrVote(string line)
{
   stringstream maStringStream(line);
   int n;
   if(voteNow==1)
     checkVote(line);
   else
   {
     if(maStringStream >> n)
     {
       checkVote(line);
       voteNow=1;
     }
     else
```

```
}

//This function checks if votes are valid
void validVoteCheck(string t) {
   int voteVoter=0;
   int incCount=0;
   stringstream stream(t);
   int n;
   while (stream >> n) {
     voteVoter++;
     tempVotes[voteVoter-1]+=n;
     incCount+=n;
   }
  if ((incCount>allowedVotes) || (voteVoter != numchoice))
    spoiledVoteCount();
  else
    readValidVotes();
}
// this function stores names into the name list.
void addchoice(string name) {
  if(numchoice<10) {
    names[numchoice]=name;
    numchoice++;
    if(isFlag==0) allowedVotes++;
  }
}

//function checks if string is name or vote
void checkNameVote(string s) {
   stringstream stream(s);
   int n;
   if(startVote==1)
     validVoteCheck(s);
   else {
     if(stream >> n) {
       validVoteCheck(s);
       startVote=1;
     }
     else
```

```
       addCandidate(line);
  }
}
/*********** readVotes *************
     Purpose: This function read inputs and call nameOrVote to let it decide whatever its a name or vote
     Returns: This function returns nothing
************************************/
void readVotes()
{
  int i;
  int voteVoter=0;
  //cin >> i;
  string s;
  getline(cin,s);
  while(!cin.eof())
  {
    nameOrVote(s);
   // cout << s << endl;
    getline(cin,s);
  }
}
/*********** printResults **********
     Purpose: This function print all the information we get from cin and count
     Returns: This function returns nothing
************************************/
void printResults()
{
  cout << "Number of voters: " << total << endl;
  cout << "Number of valid ballots: " << valid << endl;
  cout << "Number of spoilt ballots: " << spoilt << endl << endl;

  cout << left << setw(15) << "Candidate" << right << setw(3) << "Score" << endl << endl;
  for(int i=0; i<candidate; i++)
  {
    cout << left << setw(15) << nameList[i] << right << setw(3) << voteList[i] << endl;
  }
}
/*********** main ******************
     Purpose: Main
     Returns: state
************************************/
int main(int argc, char *argv[])
{
  if(argc == 2)
  {
    stringstream maArg(argv[1]);
    maArg >> voteAllow;
    argFlag=1;
  }
  readVotes();
  printResults();
  return 1;
```

```
       addchoice(s);
  }
}
//this function reads inputs and calls checkNameVote to decide if it's a name or a vote
void readVotes() {
  int voteVoter=0;
  string s;
  getline(cin,s);
  while(!cin.eof()) {
    checkNameVote(s);
    getline(cin,s);
  }
}
//this function prints all info
void printResults() {
  cout << "# of votes = " << totalVotes << endl;
  cout << "# of valid votes: " << validVotes << endl;
  cout << "# of spoiled votes: " << spoiledVote << endl << endl;
  cout << left << setw(15) << "numchoice" << right << setw(3) << "Score" << endl << endl;
  for(int i=0; i<numchoice; i++) {
    cout << left << setw(15) << names[i] << right << setw(3) << numVotes[i] << endl;
  }
}

int main(int argc, char *argv[]) {
  int result = 1;
  if(argc == 2) {
    stringstream maArg(argv[1]);
    maArg >> allowedVotes;
    isFlag = 1;
  }
  readVotes();
  printResults();
  return result;
```

```
}                                       }
```