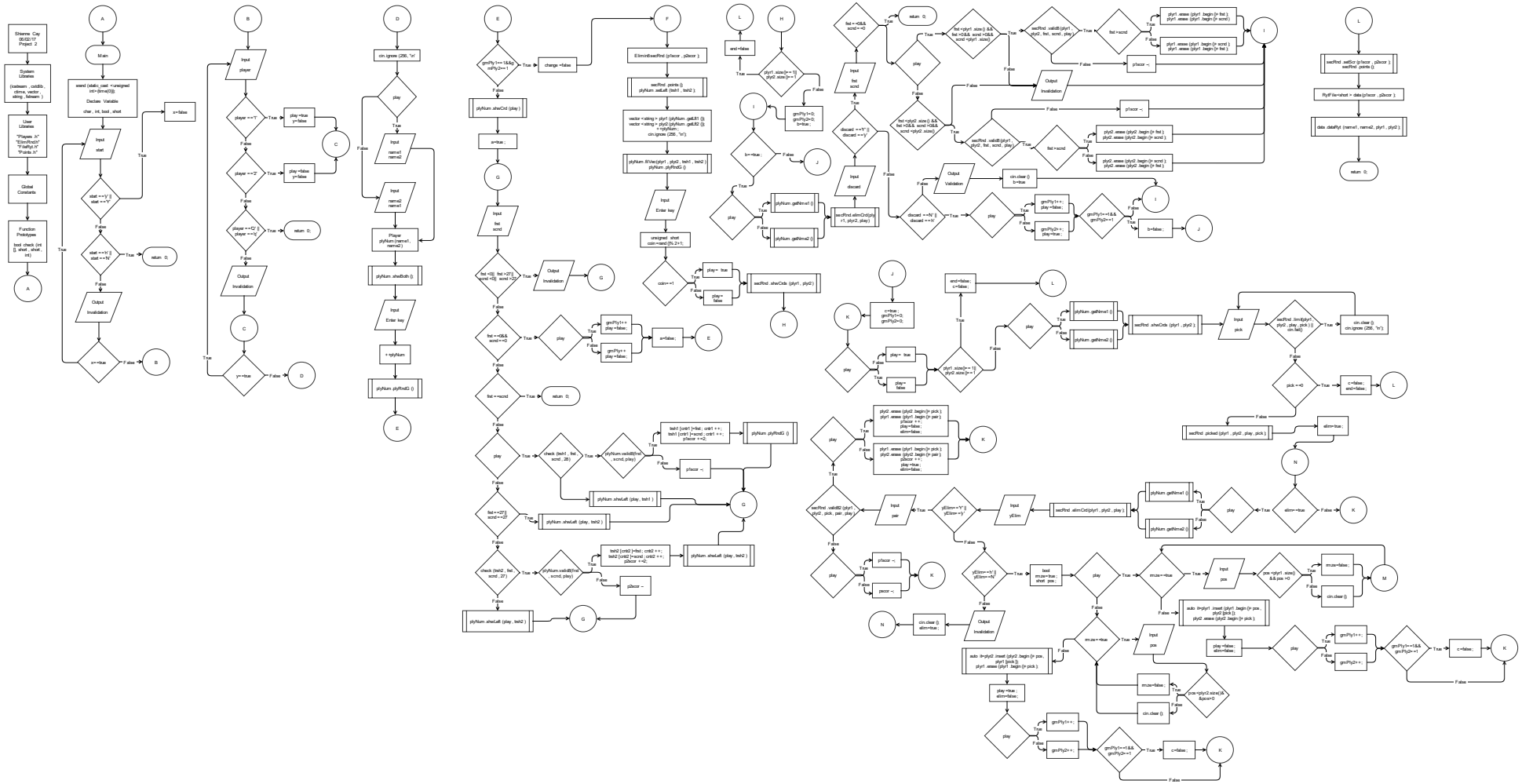
Sitterne Cay
06/02/17
Project 2
System Libraries
(iostream , cstdlib , ctime, vector , string , fstream)
User Libraries
"Players .h"
"EtcinRecdp"
"FiveBgfd.H"
"Points .h"
Global Constants
Function Prototypes
bool check (int [] , short , short , int)
Main
srand (static_cast <unsigned int>(time(0)))
Declare Variable
char , int, bool , short
Input
start
start =='Y' || start ==Y'
start ==N' || start ==N'
return 0;
Output
Invalidation
x==true
x=false
Input
player
player ==1'
player ==2'
play =true
y=false
player ==C' || player ==c'
Output
Invalidation
y==true
cin.ignore (256, '\n'.
play
Input
name1
name2
Player
plyNum (name1, name2)
plyNum .shwBoth ()
Input
Enter key
+ +plyNum
plyNum .plyRndG ()
gmPly1==1&&gmPly2==1
change =false
plyNum .shwCrd (play)
s=true ;
Input
fst
scnd
fst ==0&& scnd==0
Output
Invalidation
play
gmPly1++
play =false;
gmPly++
play =false;
s=false ;
fst ==scnd
check (tsh1 , fst, scnd , 26)
plyNum.validB(fst, scnd, play)
tsh1 [cnt1]=fst ; cnt1 ++;
tsh1 [cnt1]=scnd ; cnt1 ++;
p1scor -;
plyNum .shwLeft (play , tsh1)
fst ==27'|| scnd ==27
plyNum .shwLeft (play , tsh2)
check (tsh2 , fst, scnd , 27)
tsh2 [cnt2]=fst ; cnt2 ++;
tsh2 [cnt2]=scnd ; cnt2 ++;
p2scor +=2;
p2scor -
plyNum.shwLeft (play , tsh2)
E1stminBwsFind (p1scor , p2scor)
secfrnd .setScr (p1scor , p2scor)
secfrnd .points ()
plyNum .shwLeft (play , tsh1)
vector <string > ply1 (plyNum .getLB1 ());
vector <string > ply2 (plyNum .getLB2 ());
+ +plyNum ;
cin.ignore (256, '\n');
plyNum .fillVec(ply1 , ply2 , tsh1 , tsh2)
plyNum .plyRndG ()
Input
Enter key
unsigned short coin=rand()% 2+1;
coin==1
play= true
play= false
secfrnd .shwCrds (ply1 , ply2)
end =false
ply1 .size()==1&& ply2 .size()==1
gmPly1=0;
gmPly2=0;
b=true ;
play
plyNum .getNme1 ()
plyNum .getNme2 ()
secfrnd .elimCrd(ply1 , ply2 , play)
fst ==0&& scnd==0
return 0;
Input
fst
scnd
discard ==Y' || discard ==y'
Input
discard
fst ==ply1 .size () && fst ==0&& scnd >ply1 .size()
fst ==ply2 .size () && fst ==0&& scnd >ply2 .size()
secfrnd .validB(ply1 , ply2 , fst , scnd , play)
fst =scnd
ply1 .erase (ply1 .begin (+ fst) , ply1 .erase (ply1 .begin (+ scnd)
ply1 .erase (ply1 .begin (+ scnd) , ply1 .erase (ply1 .begin (+ fst)
p1scor -;
Output
Invalidation
ply2 .erase (ply2 .begin (+ fst) , ply2 .erase (ply2 .begin (+ scnd)
ply2 .erase (ply2 .begin (+ scnd) , ply2 .erase (ply2 .begin (+ fst)
p1scor -;
discard ==N' || discard ==n'
Output
Validation
cin.clear () ;
b=true ;
gmPly1++ ;
play =false;
gmPly2++ ;
play =true ;
gmPly1==1&& gmPly2==1
b=false ;
secfrnd .elimCrd(ply1 , ply2 , play)
secfrnd .shwCrds (ply1 , ply2)
c=true ;
gmPly1=0;
gmPly2=0;
end=false;
c=false;
Input
pick
secfrnd .lim(ply1 , ply2 , play , pick) || cin.fail()
cin.clear () ;
cin.ignore (256, '\n');
pick ==0
c=false;
end=false;
secfrnd .picked (ply1 , ply2 , play , pick)
elim=true ;
ply2 .erase (ply2 .begin (+ pick) ;
ply1 .erase (ply1 .begin (+ pair)
p1scor + +;
play =false;
elim=false;
ply1 .erase (ply1 .begin (+ pick) ;
ply2 .erase (ply2 .begin (+ pair)
p2scor + +;
play =true ;
elim=false;
secfrnd .validB2 (ply1 , ply2 , pick , pair , play)
Input
pair
yElim==Y'|| yElim==y'
Input
yElim
secfrnd .elimCrd(ply1 , ply2 , play)
p1scor -;
p2scor -;
yElim==n'|| yElim==N'
bool
m:ze=true ;
short pos ;
m:ze==true
Input
pos
pos <ply1 .size() && pos >0
m:ze=false;
cin.clear () ;
auto it=ply1 .inset (ply1 .begin (+ pos , ply2 [pick]);
ply2 .erase (ply2 .begin (+ pick) ;
play =false;
elim=false;
m:ze==true
Input
pos
pos <ply2 .size()&& &pos>0
m:ze=false;
cin.clear () ;
auto it=ply2 .inset (ply2 .begin (+ pos , ply1 [pick]);
ply1 .erase (ply1 .begin (+ pick) ;
play =true ;
elim=false;
gmPly1++ ;
gmPly2++ ;
gmPly1==1&& gmPly2==1
c=false ;
secfrnd .setScr (p1scor , p2scor)
secfrnd .points ()
RyFile=short = data (p1scor , p2scor) ;
data dataRyt (name1, name2, ply1 , ply2)
return 0;