**IPL LIVE AUCTION SYSTEM**

*By*

***Mukul Kumar : 18BEC1197***

***Abhishek Kumar : 18BEC1251***

***Samarth Raipal : 18EC1113***

A project report submitted to

**Dr. M. Braveen**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

in partial fulfilment of the requirements for the course of

**CSE2003– DATA STRUCTURE AND ALGORITHM**

in

**B.Tech. ENGINEERING**



**VIT UNIVERSITY, CHENNAI**

**Vandalur – Kelambakkam Road**

**Chennai – 600127**

# ABSTRACT

This is an LIVE IPL AUCTION platform where different teams will bid for different categories of players ( batsman , bowlers , all-rounders and wicketkeepers). Each player in every category will have a base amount from which the bidding will start. The team to bid the highest price for the player will get that player under their list while that amount will be deducted from the teams purse amount and at the same time the count of players for each team will be done . Their will be some boundary conditions for each team for the number of players and purse amount i.e. each team can bid for MAXIMUM of 22 players and will have a purse amount of 10000 Lakhs each . File Handling will be used to create and work on data of different players and general tree data structures will be used to do different operations like adding of player in a particular team, deducting purse amount , managing players , managing teams etc . Different sets of players will be made under this auction system and after completion of each set of players, a summary of the present players sold/unsold will be displayed. This auction system can be used in a very efficient way to conduct LIVE

AUCTION.

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **SERIAL NO.** |  | **NAME** | **PAGE NO.** |
| A. |  | ABSTRACT | 2 |
|  |  |  |  |
|  |  |  |  |
| 1 |  | INTRODUCTION | 4 |
|  |  |  |  |
|  | 1.1 | OBJECTIVES AND GOALS | 4 |
|  |  |  |  |
| 2 |  | MODULES | 5 |
|  |  |  |  |
| 3 |  | FLOWCHART/ARCHITECTURAL DIAGRAM | 6 |
|  |  |  |  |
| 4 |  | SOFTWARE SPECIFICATION | 7 |
|  |  |  |  |
| 5 |  | CODE IN C++ | 8 |
|  |  |  |  |
| 6 |  | OUTPUT SCREENSHOTS | 17 |
|  |  |  |  |
| 7 |  | CONCLUSION | 21 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. **INTRODUCTION**

**1.1 OBJECTIVES AND GOALS**

* **To create a LIVE AUCTION SYSTEM.**
* **To sell players to each team in an efficient manner.**
* **To Display the details of each team formed.**
* **To manage the purse of each team.**
* **To control the auction system in a computerized manner.**

# 2. MODULES

**DATA STRUCTURE USED : GENERAL TREE**

* **NODES OF TEAMS** : To manage players of each team in a generalized tree form .

* **PURSE DETAILS** : To manage purse money of each team.

* **CATEGORY OF PLAYERS** : To manage batsman , bowlers , all-rounders and wicketkeepers.

* **LIST OF UNSOLD PLAYERS** : To manage the unsold players.

* **DISPLAY** : To display player details with their respective base prices .

* **BOUNDARY CONDITIONS** : Maximum number of players (22) and purse limit.

* **BIDDIDNG SYSTEM** : To conduct the bidding process , one by one in an efficient manner.

* **DISPLAY SET** : To display the summary of a particular set of auctioned players.

**3. FLOWCHART/ARCHITECTURAL DIAGRAM**

## PLAYERS

**BATSM**

**E**

**N**

**S**

**BOWLERS**

**ALL**

**-**

**ROUNDERS**

**SOLD**

**UNSOLD**

**CHECK BOUNDARY**

**CONDITIONS**

**DEDUCT SELLING**

**AMOUNT FROM**

**PURSE AMOUNT**

**ADD THE PLAYER**

**TO THE UNSOLD**

**PLAYER LIST**

**WICKETKEEPERS**

**I**

**NCREASE THE**

**COUNT OF UNSOLD**

**PLAYERS**

**BY**

**1**

**BIDDING**

**A**

**MONG TEAMS**

**CREDIT THE PLAYER TO TEAM**

**WITH HIGHEST BID**

**FINALL**

**Y INCREASE THE**

**COUNT OF PLAYER IN THAT**

**TEAM BY 1**

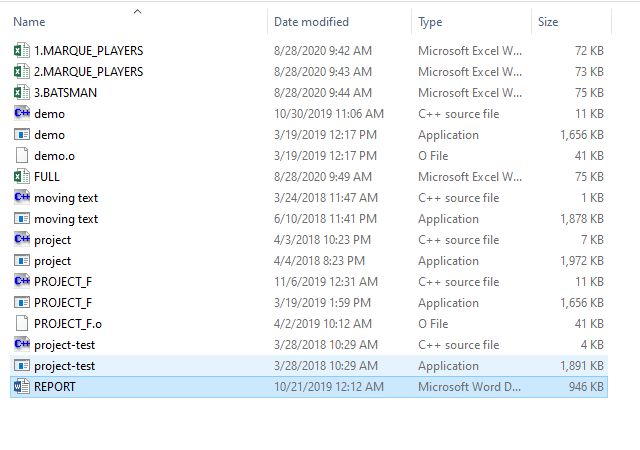
1. **SOFTWARE SPECIFICATION**

**The programming language used to write this code is C++.**

**The Software used to write the code for “LIVE IPL AUCTION SYSTEM” is**



**DEV C++ 5.11 .**



*Figure 1*

*Project Folder*

1. **C++ CODE FOR “ LIVE IPL AUCTION SYSTEM ”**

#include <bits/stdc++.h> using namespace std; typedef struct node

{ string data; struct node \*a; struct node \*b; struct node \*c;

struct node \*d;

}Node;

int cnt(float bid[8])

{

int i,c=0;

for(i=0;i<8;i++)

if(bid[i]==0)

c++;

return c;

}

void batsman(struct node \*root)

{

int c=0;

if (root != NULL)

{

cout<<root->data<<endl; if(c==0)cout<<"================\n";

batsman(root->a);

}

return;

}

void bowler(struct node \*root)

{

int c=0;

if (root != NULL)

{

cout<<root->data<<endl; if(c==0)cout<<"==============\n"; bowler(root->a);

}

return;

}

void allrounder(struct node \*root)

{

int c=0;

if (root != NULL)

{

cout<<root->data<<endl;

if(c==0)cout<<"===================\n";

allrounder(root->a);

}

return;

}

void wicketkeeper(struct node \*root)

{

int c=0;

if (root != NULL)

{

cout<<root->data<<endl;

if(c==0)cout<<"===================\n"; wicketkeeper(root->a);

}

return;

}

Node \*csk,\*kkr,\*mi,\*rcb,\*kxip,\*srh,\*rr,\*unsold,\*dd=NULL;

float

cp=10000,kp=10000,mp=10000,rp=10000,pp=10000,dp=10000,hp=10000,bp=10000; int ncp=0,nkp=0,nmp=0,nrp=0,npp=0,ndp=0,nhp=0,nbp=0,nunsold=0; int equal(string s1, string s2)

{

if((s1.compare(s2)) == 0)

return 1; else return 0;

}

Node \*newNode(string item)

{

node \*temp=new Node;

temp->data=item;

temp->a=temp->b=temp->c=temp->d=NULL; return temp;

}

Node \*newNode1(string item)

{

node \*temp=new Node;

temp->data=item; temp->a=NULL; return temp;

}

void team(Node \*t)

{

t->a=newNode("BATSMAN"); t->b=newNode("BOWLER"); t->c=newNode("ALL-ROUNDER"); t->d=newNode("WICKETKEEPER");

}

void add(Node \*head,string name)

{

if(head->a==NULL)

{

head->a=newNode1(name);

return;

}

add(head->a,name);

} int count=1;

Node\* insert(Node\* head, string name,string category)

{

Node \*temp=new Node; temp=head;

if (head == NULL) return newNode("1"); else if (equal(category,head->a->data)==1) add(head->a,name); else if (equal(category,head->b->data)==1) add(head->b,name); else if (equal(category,head->c->data)==1) add(head->c,name); else if (equal(category,head->d->data)==1) add(head->d,name); return temp; delete(temp);

}

void push(string s,string y,string g)

{

if(s=="CSK") csk=insert(csk,y,g); else if(s=="KKR") kkr=insert(kkr,y,g); else if(s=="MI") mi=insert(mi,y,g); else if(s=="RCB") rcb=insert(rcb,y,g); else if(s=="RR") rr=insert(rr,y,g); else if(s=="DD") dd=insert(dd,y,g); else if(s=="SRH") srh=insert(srh,y,g); else if(s=="KXIP") kxip=insert(kxip,y,g); else if(s=="UNSOLD")

unsold=insert(unsold,y,g);

}

void details(string a,string b,string c,string d,string e,string f,string g,string h)

{

cout<<"\n+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"<<endl;

cout<<"\n LIVE IPL AUCTION 2018

\n";

cout<<"\n+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"<<endl;

cout<<"\nPLAYER DETAILS : \n==============\n\nType : "<<b<<"\n"<<"SET

NUMBER : "<<c<<"\n"<<"Player Number : "<<a<<"\n"<<"Name : "<<d<<"

"<<e<<"\n"<<"Country : "<<f<<"\n"<<"Speciality : "<<g<<"\n"<<"Base Price : "<<h<<"\n"; cout<<"\n+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"<<endl; cout<<"\nCurrent available purse : \n------------------------\n"; cout<<"|CSK : "<<cp<<"L|KKR : "<<kp<<"L|MI: "<<mp<<"L|RCB :

"<<bp<<"L|RR : "<<rp<<"L|DD : "<<dp<<"L|SRH : "<<hp<<"L|KXIP :

"<<pp<<"L|\n"<<endl;

cout<<"+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"<<endl;

cout<<"\nCurrent Number of Players : \n--------------------------\n";

cout<<"|CSK : "<<ncp<<"|KKR : "<<nkp<<"|MI: "<<nmp<<"|RCB : "<<nbp<<"|RR

: "<<nrp<<"|DD : "<<ndp<<"|SRH : "<<nhp<<"|KXIP : "<<npp<<"|\n\nNUMBER OF

UNSOLD PLAYERS : "<<nunsold<<"\n"<<endl;

cout<<"+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"<<endl;

cout<<"\nRULES : \n======\n\n-> TO BID ENTER 1.\n-> TO DENY ENTER ANY OTHER DIGIT (PREFERABLY 0).\n\n-> The Increment in BID per bidding are as follow :\n\nCURRENT BID RANGE INCREMENT IN BID\n================= =================\n20L-95L +5L\n100L-195L +20L\n200L395L +50L\n400L AND ABOVE +100L\n\n\n+-+-+-+-+-+-+-+-+-

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-++-+-+-+\n";

}

void run\_with(const char\* filename)

{

string pteam[8]={"CSK","KKR","MI","RCB","RR","DD","SRH","KXIP"}; int i,t,cc=1,index=0,count=1; system("CLS");

ifstream ip(filename); float price,bidd,k;

string a,b,c,d,e,f,g,h,team,y;

while(!ip.eof())

{

stringstream ss; count=1;

getline(ip,a,','); getline(ip,b,','); getline(ip,c,','); getline(ip,d,','); getline(ip,e,','); getline(ip,f,','); getline(ip,g,',');

getline(ip,h,'\n');

ss<<h;ss>>bidd; k=bidd;

while(count!=0)

{

float bid[8]={1,1,1,1,1,1,1,1},u=0;

count=0; cc=1; bidd=k;

while(cc<7)

{

details(a,b,c,d,e,f,g,h);

cc=0;

for(i=0;i<8;i++)

{

if(bid[i]!=0)

{

cout<<pteam[i]<<" : "; cin>>bid[i]; if(bid[i]==1)

{

if(bidd>=20 && bidd<100)

bidd=bidd+5;

else if(bidd>=100 && bidd<200)

bidd=bidd+20;

else if(bidd>=200 && bidd<400)

bidd=bidd+50;

else

bidd=bidd+100;

index=i; u++; }

else

bid[i]=0;

cout<<"CURRENT BID : "<<bidd<<endl;

}

}

cc=cnt(bid);

system("CLS");

}

if(cc==7 || u!=0)

{

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | cout<<pteam[index]; | |
|  |  | if(pteam[index]=="CSK" && cp>=20 && (cp-bidd)>=0 && ncp<=22) | |
|  |  | { cp=cp-bidd;ncp++;} | |
| nkp<=22) |  | else if(pteam[index]=="KKR" && kp>=20 && (kp-bidd)>=0 && | |
|  |  | { kp=kp-bidd;nkp++;} | |
| nmp<=22) |  | else if(pteam[index]=="MI" && mp>=20 && (mp-bidd)>=0 && | |
|  |  | { mp=mp-bidd;nmp++;} | |
| nbp<=22) |  | else if(pteam[index]=="RCB" && bp>=20&& (bp-bidd)>=0 && | |
|  |  | { bp=bp-bidd;nbp++;} | |
| nrp<=22) |  | else if(pteam[index]=="RR" && rp>=20 && (rp-bidd)>=0 && |
|  |  | { rp=rp-bidd;nrp++;} |
| ndp<=22) |  | else if(pteam[index]=="DD" && dp>=20 && (dp-bidd)>=0 && |
|  |  | { dp=dp-bidd;ndp++;} |
| nhp<=22) |  | else if(pteam[index]=="SRH" && hp>=20 && (hp-bidd)>=0 && |
|  |  | { hp=hp-bidd;nhp++;} |
| npp<=22) |  | else if(pteam[index]=="KXIP" && pp>=20 && (pp-bidd)>=0 && |
|  |  | { pp=pp-bidd;npp++;} |
|  |  | else |
|  |  | { |
| team.\n\n"; |  | cout<<"\nPurse Limit Exceeded !!\nTry again with another |
|  |  | count++; |
|  |  | } |

}

else if(cc==8)

{

y=d+" "+e;

push("UNSOLD",y,g); nunsold++;}}

if(cc==7)

{

y=d+" "+e;

push(pteam[index],y,g);} system("CLS");}

cout<<"==========================\n"; cout<<"Summary of PREVIOUS SET : "<<endl; cout<<"==========================\n"; cout<<"==========================\n";

cout<<csk->data<<" : \n=========================="<<endl; batsman(csk->a);cout<<"\n"; bowler(csk->b);cout<<"\n"; allrounder(csk->c);cout<<"\n";

wicketkeeper(csk->d);cout<<"\n";

cout<<"==========================\n";

cout<<kkr->data<<" : \n=========================="<<endl; batsman(kkr->a);cout<<"\n"; bowler(kkr->b);cout<<"\n";

allrounder(kkr->c);cout<<"\n";

wicketkeeper(kkr->d);cout<<"\n";

cout<<"==========================\n";

cout<<mi->data<<" : \n=========================="<<endl; batsman(mi->a);cout<<"\n"; bowler(mi->b);cout<<"\n"; allrounder(mi->c);cout<<"\n";

wicketkeeper(mi->d);cout<<"\n";

cout<<"==========================\n";

cout<<dd->data<<" : \n=========================="<<endl; batsman(dd->a);cout<<"\n"; bowler(dd->b);cout<<"\n"; allrounder(dd->c);cout<<"\n"; wicketkeeper(dd->d);cout<<"\n";

cout<<"==========================\n";

cout<<rcb->data<<" : \n=========================="<<endl; batsman(rcb->a);cout<<"\n"; bowler(rcb->b);cout<<"\n"; allrounder(rcb->c);cout<<"\n";

wicketkeeper(rcb->d);cout<<"\n";

cout<<"==========================\n";

cout<<rr->data<<" : \n=========================="<<endl; batsman(rr->a);cout<<"\n"; bowler(rr->b);cout<<"\n"; allrounder(rr->c);cout<<"\n";

wicketkeeper(rr->d);cout<<"\n";

cout<<"==========================\n";

cout<<kxip->data<<" : \n=========================="<<endl; batsman(kxip->a);cout<<"\n"; bowler(kxip->b);cout<<"\n"; allrounder(kxip->c);cout<<"\n"; wicketkeeper(kxip->d);cout<<"\n";

cout<<"==========================\n";

cout<<srh->data<<" : \n=========================="<<endl; batsman(srh->a);cout<<"\n"; bowler(srh->b);cout<<"\n"; allrounder(srh->c);cout<<"\n";

wicketkeeper(srh->d);cout<<"\n"; cout<<"==========================\n";

cout<<unsold->data<<" : \n=========================="<<endl; batsman(unsold->a);cout<<"\n"; bowler(unsold->b);cout<<"\n"; allrounder(unsold->c);cout<<"\n";

wicketkeeper(unsold->d);cout<<"\n"; cout<<"\n==========================";

ip.close(); } int main(){

int r;

cout<<"=======================================================

=====================================\n"; cout<<" WELCOME TO LIVE AUCTION

PLATFORM\n=======================================================

====================================="; cout<<"\nThis is an INDIAN PREMIER LEAGUE PLATFORM where you can bid for your favourite players.\n\nThere are a set of 52 categorised players each containing a set of certain...\nMARQUE PLAYERS , BATSMANS , WICKETKEEPERS , BOWLERS & ALL-ROUNDERS.\n\nKeep bidding for players of your choice and do keep an eye on your remaining purse.\n\n\nRULES :\n=====\n\n-> Enter 1 to BID for the given Player.\n-> Enter any other number(preferably 0) to DENY.\n";

cout<<"\n======================================================

======================================\nPRESS 1 to begin the auction...\n"; cin>>r;

if(r==1){

csk=newNode("Chennai Super Kings");kkr=newNode("Kolkata Knight

Riders");mi=newNode("Mumbai Indians");rcb=newNode("Royal Challengers Bangalore");

rr=newNode("Rajasthan Royals");dd=newNode("Delhi

Daredevils");kxip=newNode("Kings XI Punjab");srh=newNode("Sunrisers

Hyderabad");unsold=newNode("UNSOLD");

team(csk);team(kkr);team(mi);team(rcb);team(rr);team(dd);team(kxip);team(srh),tea m(unsold);

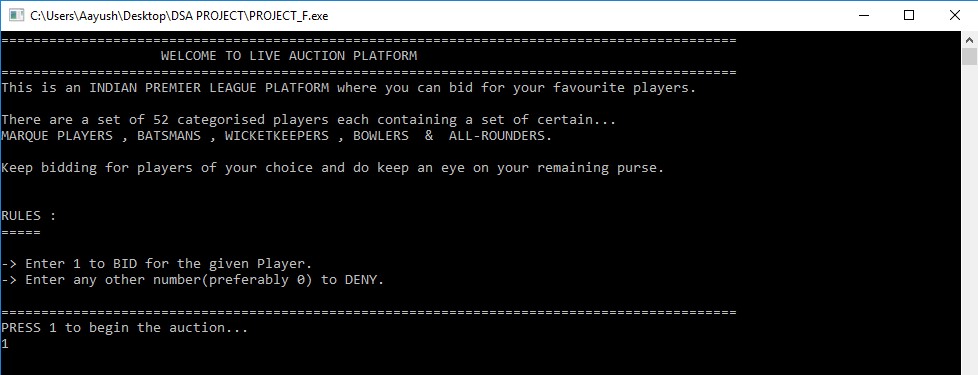
int j,z=1;

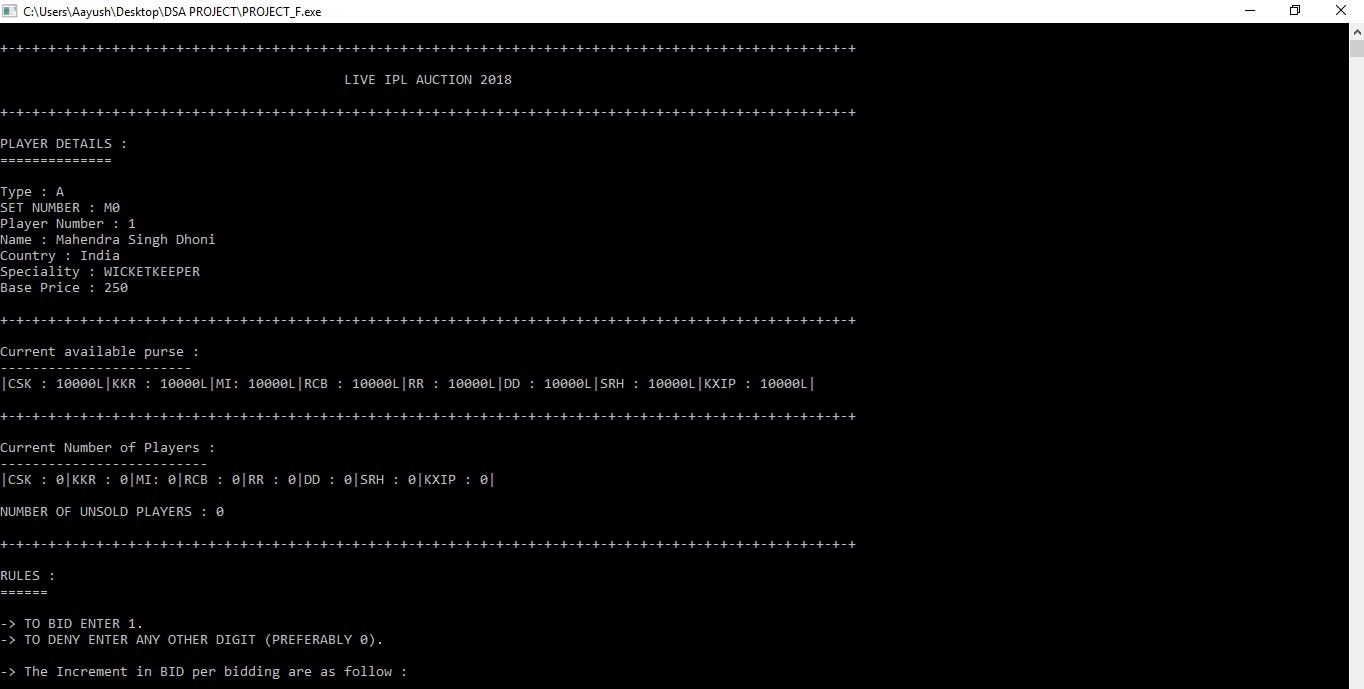
char ll[][20]={"D:\\Book1.csv","D:\\Book2.csv","D:\\Book3.csv"}; for(j=0;j<3;j++){ if(z==1) run\_with(ll[j]); else

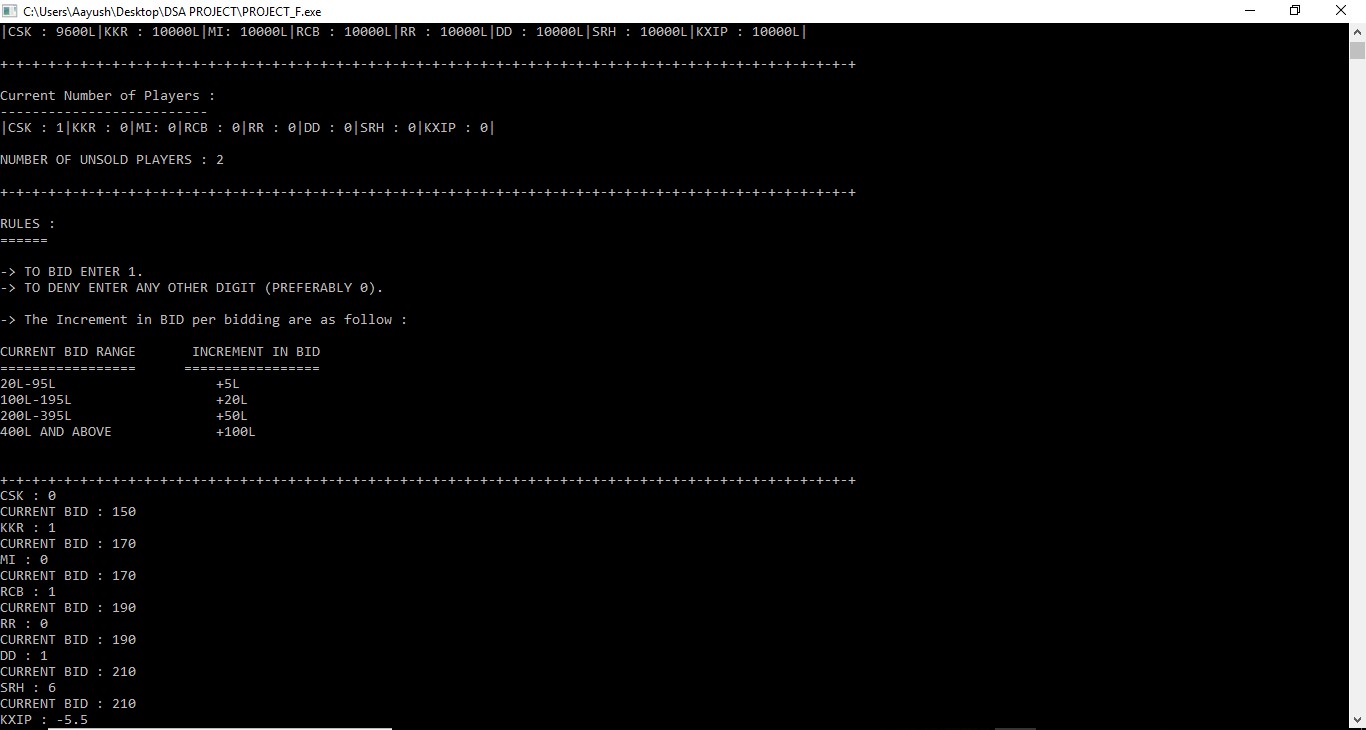
cout<<"Try Again : ";

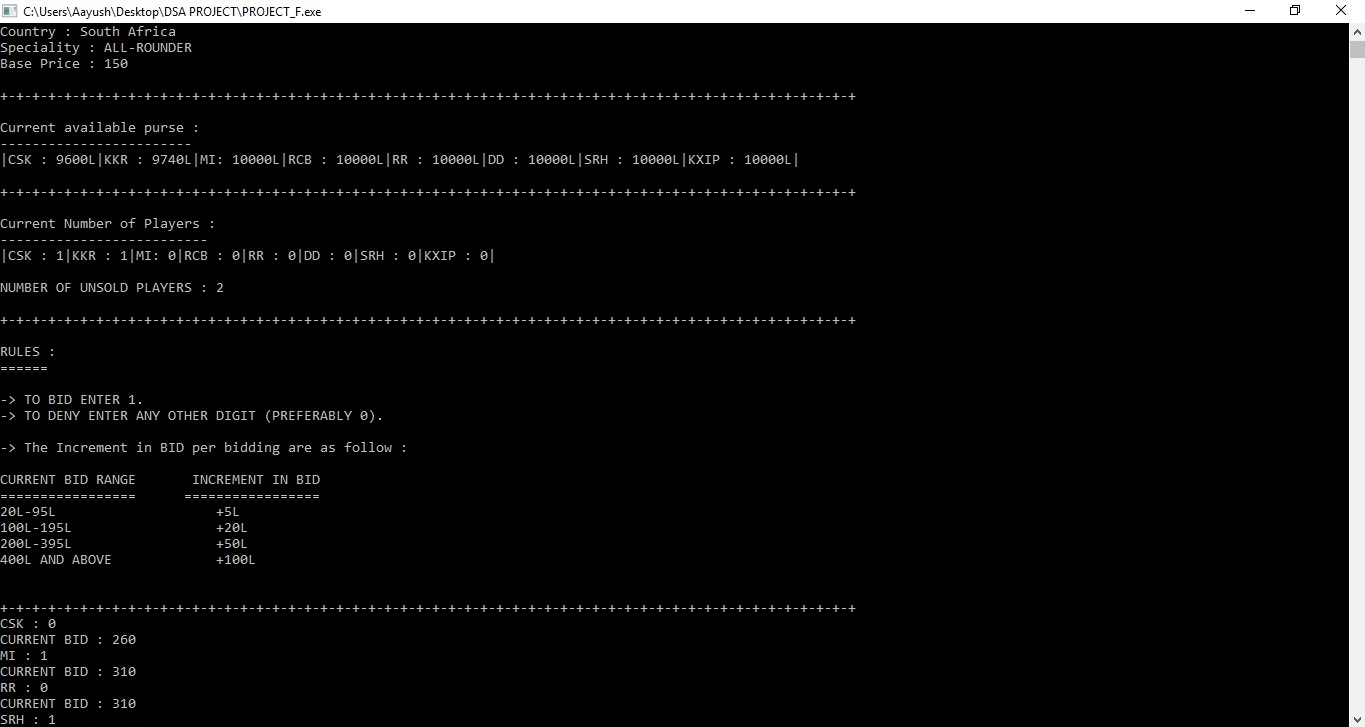
cout<<"\nEnter 1 to continue with another set : "; cin>>z; } }}

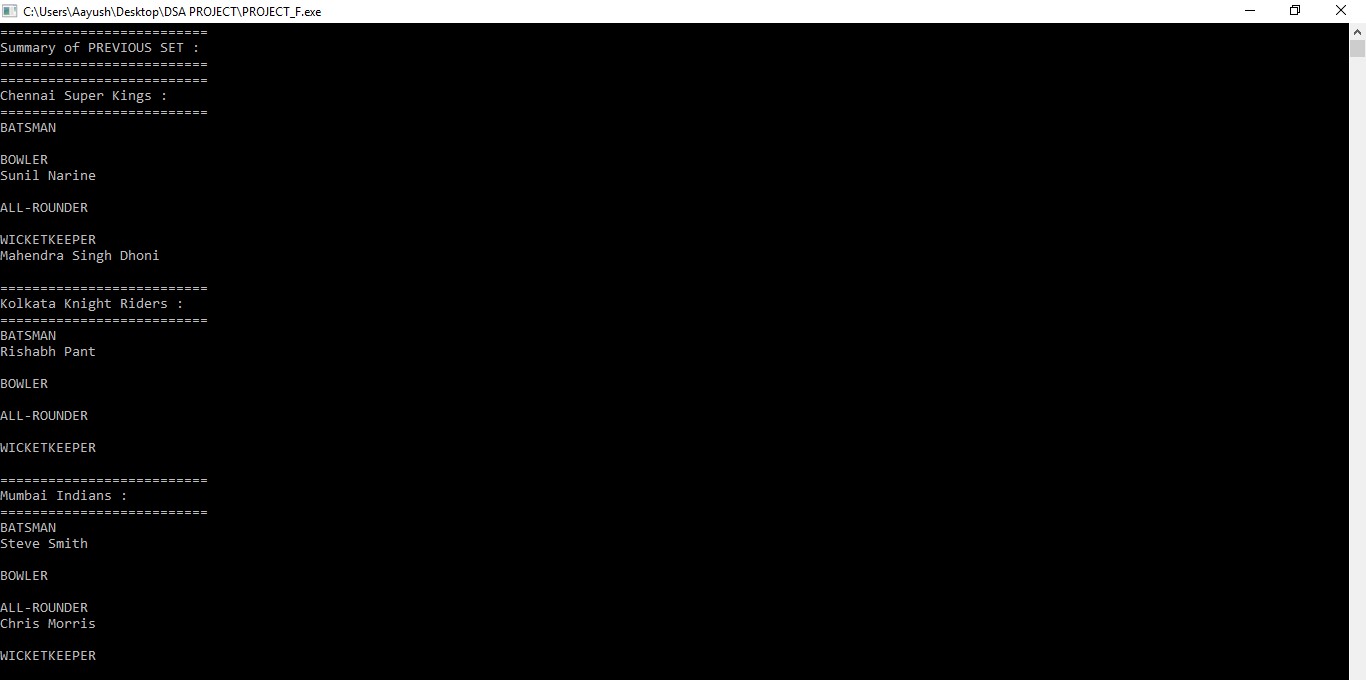
### 6. OUTPUT SCREENSHOTS

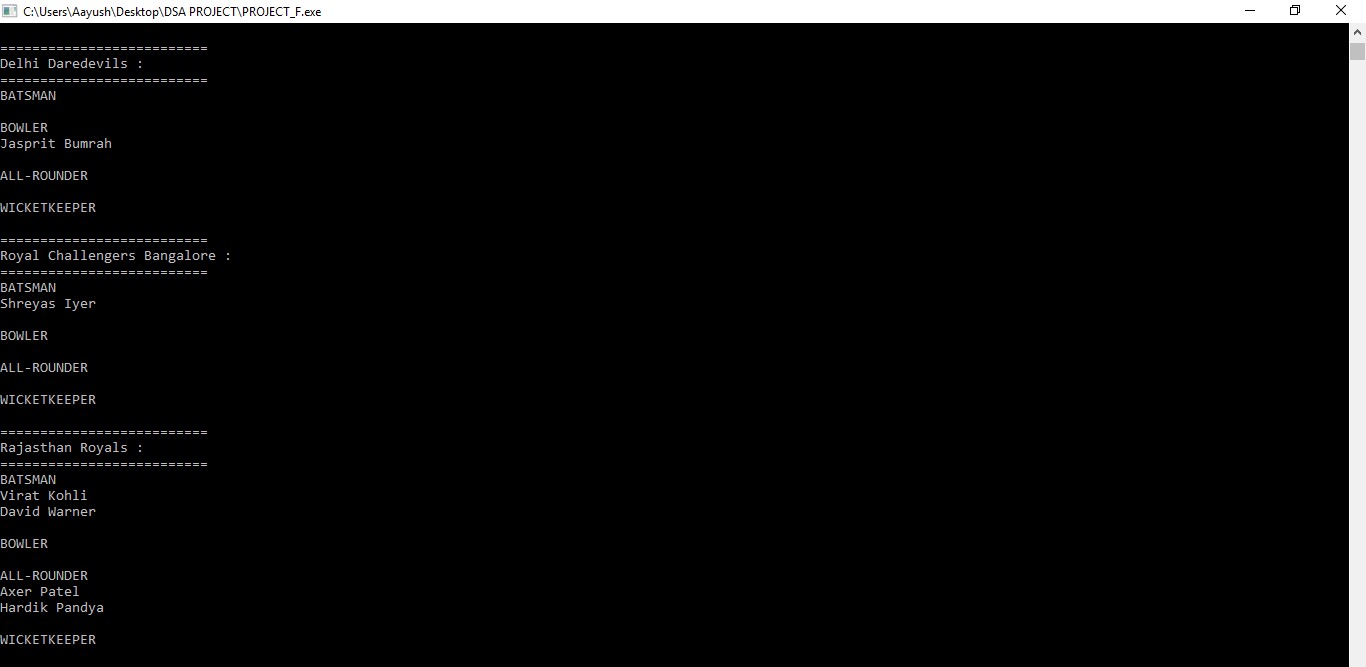


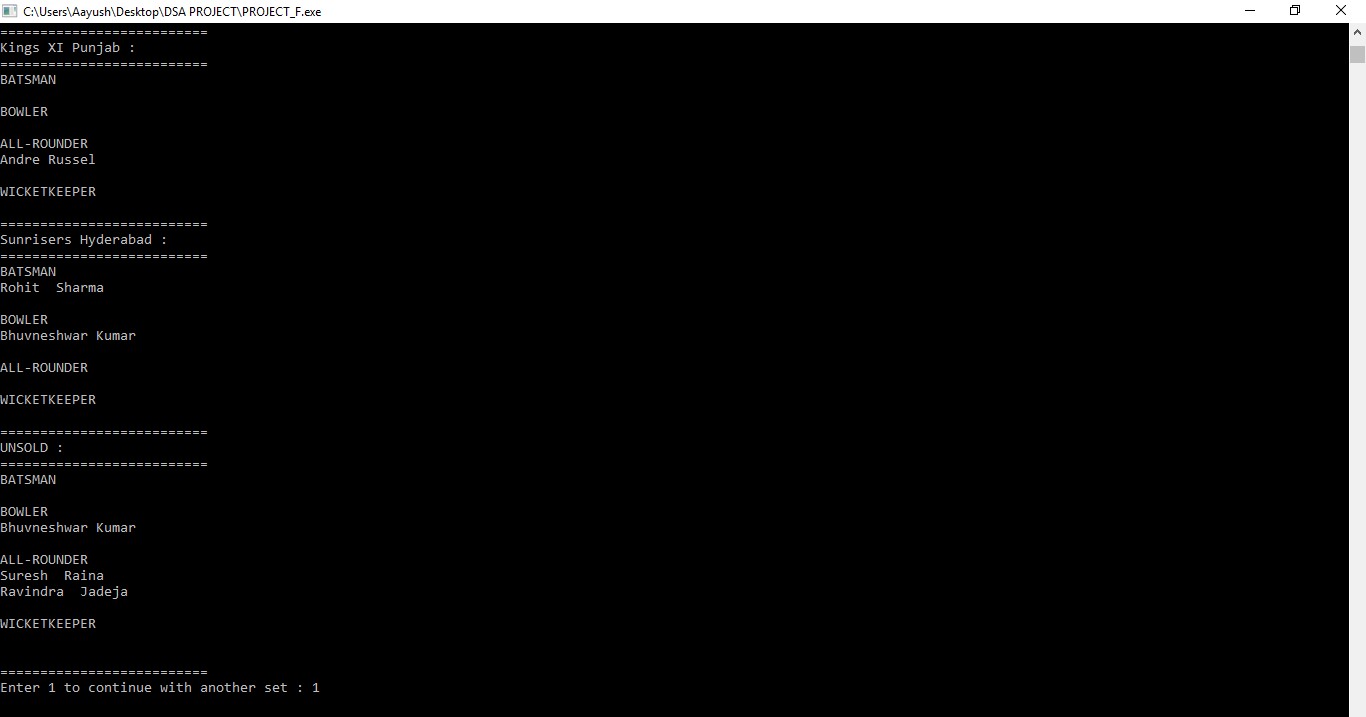


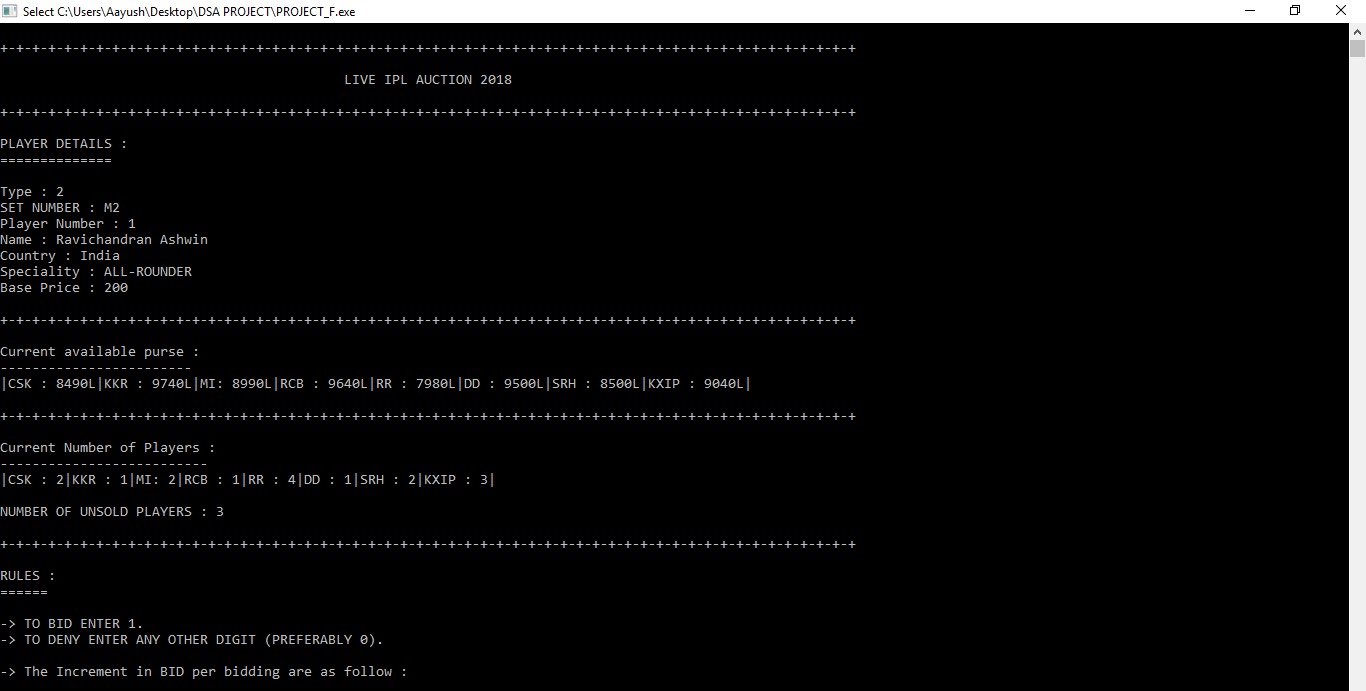












**7 . CONCLUSION**

**This “IPL LIVE AUCTION SYSTEM” will help all the auctioneers all over the world to conduct a smooth LIVE AUCTION.**

**This is a simple C++ coded “ LIVE IPL AUCTION SYSTEM” to**

**SELL IPL players to the respective playing teams in an effective and efficient manner.**

**General Tree Data Structure is used here to do all the operations .**

**It is fast , compact , efficient and very effective AUCTION SYSTEM .**