# Hand

- hand[20]: Card

// Member Functions

- +~Hand():
- + setHand(&d:Deck, i:int): void
- +getHand(x:int):Card
- + Bet():

- bet: float - amount: float

-\*lost: float

- + Bet(b: float, a: float):
- + setBet(b:float): void + setAmount(a:float): void
- + getBet():float + getAmount():float
- + getLost(index:int):float
- + PlaceBets(p\*:Player, numOfP:int):void

Bet

# Player

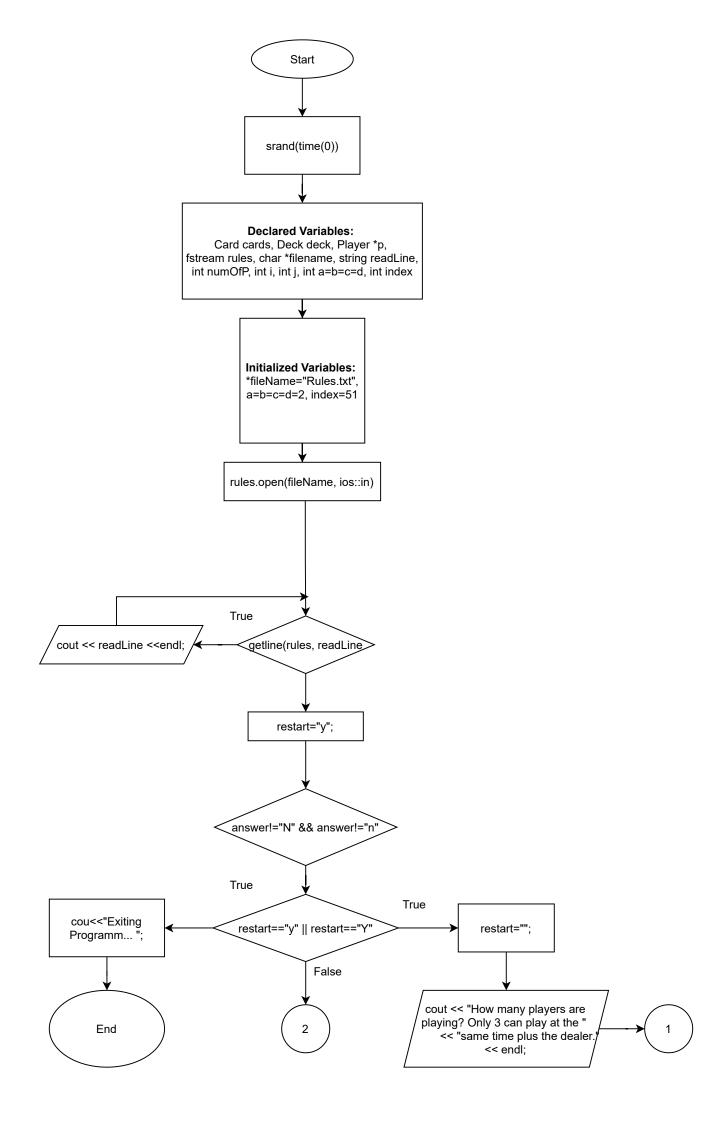
- name:string
- total:int
- credit:float
- + Player():
- + Player(n:string):
- + setName(n:string): void
- + setTotal(\*l:Hand , x : int):void
- + setCredit(c:float): void
- + getName():string + getTotal():int
- + getCredit():float
- + Player (&p:Player, c:float): friend operator+=

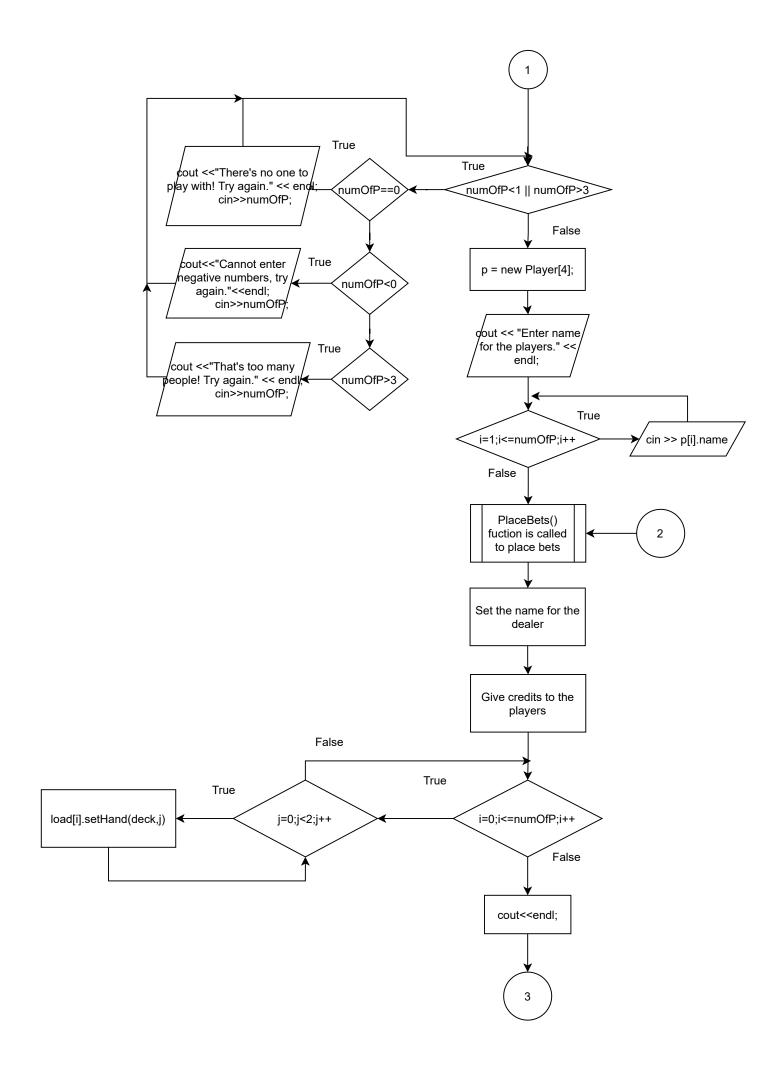
### Deck

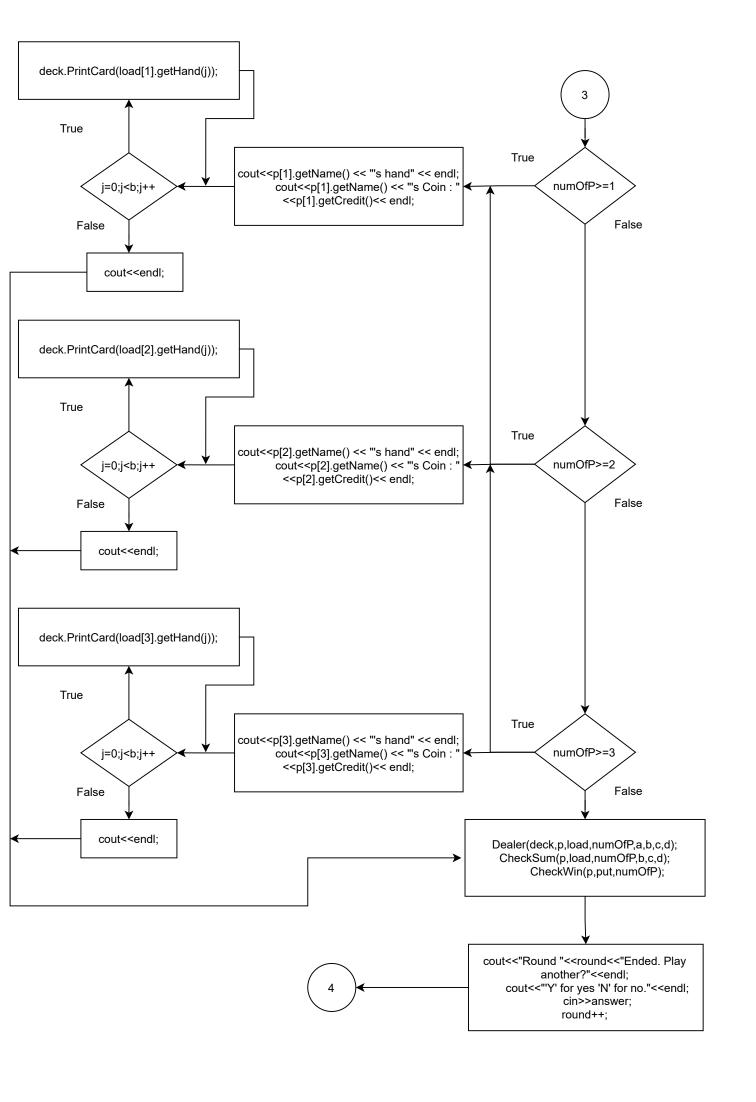
- arrCards[52]:Card
- index:int
- + Deck():
- + setArrayOfCards(c:Card):void +getArrayOfCards(): const Card
- + PrintCard(c:Card):void
- + PrintValue(c:Card):void
- + PrintSuit(c:Card):void
- + shuffle():void + PrintAll():void
- + draw():Čard

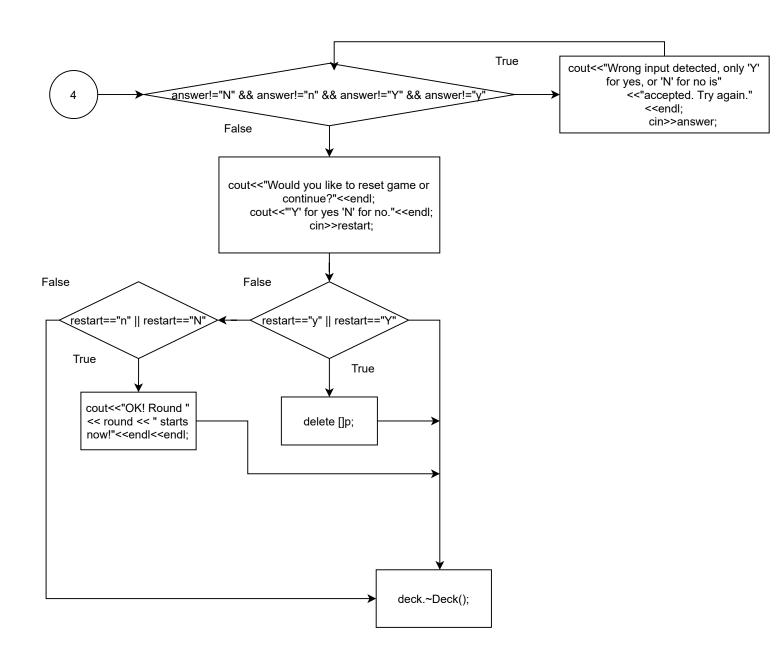
# Card

- + name:CardName
- + suit:Suits
- + value:int









void Dealer(Deck &deck, Player \*p, Hand \*I, int numOfP, int &index,int &a, int &b, int &c, int &d) int choice,i,j; True out << "Press 1 to let numOfP>=1 player one draw."< False True cout << "Press 2 to let numOfP>=2 player two draw."< True False cout << "Press 3 to let numOfP>=3 player three draw."< False cout <<"Press 4 to continue."<> choice; 2 choice!=4 False 3 False False choice==1 choice==2 choice==3 True True True p[1].hand[b] =p[2].hand[c] = p[3].hand[d] =draw(deck, index); draw(deck, index); draw(deck, index); b++; C++; d++;

