|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| Y/N for playTime  Y/N for game start with previous score  userInput on main menu  numSticks  numPickUp  playGame Y/N  rollThirdDice Y/N | Processing Items:   1. TotalEP 2. gameEP 3. playTime 4. gamesWon 5. gamesLost   Algorithm:  **MAIN**  displayWelcome();  bool playtime = getPlayTime();  while (playtime = true)  Ask for new game or return to last (Y/N)  If user wants to return previous score (Y or y)  totalEP = score on file  display totalEP  Else  totalEP = 200  display totalEP  While (totalEP < 300 or > 0)  gameEP = 0  store totalEP on file  gameSelection = displayMainMenu();  if gameSelection = 1  displaySticksRules();  bool playGame = true  While (playGame = true)  gameEP = playStickGame();  totalEP = totalEP + gameEP  getScoreBoard(totalEP, childWinCount, userWinCount, gameEP);  playGame = getReplayGame()  END WHILE  else if gameSelection = 2  displayMotherRules();  bool playGame = true  While (playGame = true)  gameEP = playMotherGame();  totalEP = totalEP + gameEP  getScoreBoard(totalEP, childWinCount, userWinCount, gameEP);  playGame = getReplayGame()  END WHILE  else If gameSelection = 3  display game has restarted  totalEP = 200  else if gameSelect = 4  Display “Score has been saved. Press any key to leave.”  Console.ReadKey();  break;  else  display invalid input, please select a valid menu option or quit  END WHILE  If totalEP >= 300  The child won, you failed your babysitting duties!  Else if totalEP <= 0  You won, you are a successful babysitter!  Else  Thanks for playing, see you when you get back!  playtime = getContinuePlayTime();  END WHILE  Display goodbye  Console.ReadKey();  **METHODS**:  displayWelcome()  display game instructions  getPlayTime();  Ask user if they want to play  Read in user input  If user input = y or Y  Return true  Else  Return false  getContinuePlayTime();  Ask user if they want to keep going or are they SURE they want to quit the game (and save progress)  Read in userInput  If user input is y or Y  Return true  Else  Return false  displayMainMenu();  Ask user to selection from menu:  1 for Sticks  2 for Mother May I  3 to restart the game  4 to quit the game  gameSelection = Read in user input  return gameSelection  playStickGame();  Ask for numSticks between 20 and 100  numSticks = int.Parse(console.ReadLine())  while numSticks < 20 or >100  Display not a valid entry  Ask for numSticks between 20 and 100  numSticks = int.Parse(console.ReadLine())  END WHILE  sitterPoints = 0  childPoints = 0  While (numSticks > 0)  Ask for NumPickedUp (number of sticks to pick up 1-3 with default being 1)  numPickedUp = int.Parse(console.readline())  Switch numPickedUp  Case 1:  numSticks = numSticks – 1  childPoints = childPoints + 1  Break;  Case 2:  numSticks = numSticks – 2  childPoints = childPoints + 2  Break;  Case 3:  numSticks = numSticks – 3  childPoints = childPoints + 1  Break;  Default  numSticks = numSticks – 1  childPoints = childPoints + 1  if (numSticks > 0)  generateRandomNum(1,3)  numSticks = numSticks – randomNum  sitterPoints = sitterPoints + randomNum  if (numSticks <= 0)  Display child wins  gameEP = childPoints  else  Display babysitter wins  gameEP = -(sitterPoints)  END WHILE  return gameEP;  getMotherGame() (NOT FINISHED, STILL WORKING OUT LOGIC)  Ask user to press any key to roll  Console.ReadKey()  sitterFirstRoll = randomNum generator (0, 9)  Display sitterFirstRoll  Ask user to press any key to roll second dice  sitterSecondRoll = randomNum generator (0, 9)  Display sitterSecondRoll  sitterTotalRoll = sitterSecondRoll + sitterFirstRoll  Display sitterTotalRoll  Ask user if they would like to roll the third dice (Y/N)  rollThirdDice = char.Parse(console.ReadLine())  If rollThirdDice = y or Y  While rollThirdDice = y or Y  sitterThirdRoll = randomNum generator (0, 9)  sitterRollTotal = sitterRollTotal + sitterThirdRoll  Ask user if they would like to roll the third dice (Y/N)  rollThirdDice = char.Parse(console.ReadLine())  END WHILE  else  getScoreBoard(totalEP, gamesLost, gamesWon, gameEP);  if gameEP >0  gamesLost++  Display “The child gained “ + gameEP + “EP from this game. Oh no!”  else  gamesWon++  Display “The child lost “ + gameEP + “ EP from this game. Good work!”  Display “You have won” + gamesWon + “games”  Display “You have lost “ + gamesLost + “ games”  Display “The current total EP is: “ + totalEP  getReplayGame()  Ask if user wants to play game again Y or N?  If Y or y  Return true  Else  Return false | totalEP  gamesWon  gamesLost |