Project 2

Title

Guessing Game
with Dice

Course

CIS-5

Section

41596

Due Date

12 February 2023

Author

Dana Rodriguez

Table of Contents

1 Introduction

A simple guessing game but with dice. This can help those who do not have physical dice around but would like to play a game with chance. The game's simplicity can be played by people of all ages, as long as they can read and count. The game's visuals also help see the number of dice rolled. It also includes different sorting algorithms at the end to show the dices and sum you had.

2 Game Play and Rules

Start by running the game. The title of the game should appear along with a message about how to play the game. This will allow the players to know the rules and input their name. This will start the game by picking their numbered targets from 1-21.

```
        Guessing game with dice
Pick your targeted amount
Proceed to roll as any dice to get your targeted amount
What is your name?
dana
What number do you want to target to from 1-21?
12
```

Once entered, you are asked if you want to roll one or two dice. After the input of how many rolls, a visual of how many dice were picked is shown and accounted for. It first tells you the total of your rolls and then the difference between what you rolled and the targeted amount. This can help with knowing how much more is left and allow the player to think to move either one or two more dice left.

```
2
=============
| ●       ● |
|     ●     |
| ●       ● |
=============

=============
| ●       ● |
| ●       ● |
| ●       ● |
=============
You total amount so far is 11
The difference between what you rolled and your target number is:  1
Do you want to roll 1 or 2 dice?
```

Depending on how much you have left, it will continue to ask how many dice you would like to roll for the chance of winning. If rolled and your total is more than what you targeted, the difference would be in the negatives and would message you lost. If rolled and the total is what your target is, it will tell you, you won.

```
=============
|           |
|     ●     |
|           |
=============

You total amount so far is 12
The difference between what you rolled and your target number is:  0
You won !!! :D
Do you want to play again? y/n
```

```
Do you want to roll 1 or 2 dice? 
1
=============
| ●         |
|     ●     |
|         ● |
=============

You total amount so far is 3
The difference between what you rolled and your target number is:  -2
You lost! Sorry  :(
Do you want to play again? y/n
```

After each game, it will ask if you would like to play again and you have the choice to answer. If yes, it will message out “Let's go!”, it will ask again for your target number. If not, it will say thank you for playing and show your bubble sort and selection sort with the numbers used within the dice game.

```
Do you want to play again? y/n
n
Thank you for playing !!!!
Bubble sort dice rolls:
3 11 12
Selection sort dice rolls:
3 11 12
```

3 Development Summary

Lines of Code	461
Comment Line	89
Blank Lines (White Spaces)	29
Total Lines of File	373

I used NetBeans to run my code. There are four different versions of this game as it progresses from mistakes from each version. There are many mistakes as it comes from one function to having many different prototypes and algorithms. I tried my best with the allotted time and chose what was needed within the project to work and not get off the rails with different concepts.

2.1.0

With having to change my original game to a game with functions and arrays, I first started to add the different prototypes to start up the game. This took the most time since the while loop in the OG game makes everything flow easier, breaking it up was hard to split within different prototypes. To make sure each prototype was working, I added a few sentences to be displayed if the program was running as it supposes to be.

```
Do you want to roll 1 or 2 dice? 
2
rolls2: 0
=============
| ●       ● |
| ●       ● |
| ●       ● |
=============
sumd216:6

============
| ●     ● |
|         |
| ●     ● |
============
sumd224:10
sum:10
sumafter2roll:10
```

This help with finding what needs to be fixed within the program before continuing with adding more concepts. What was bugging out is that the game would fail after one roll.

```
You total amount so far is 10
The difference between what you rolled and your target number is:  11
value of play:0

RUN SUCCESSFUL (total time: 1m 12s)
```

It also continues to fail after you lost and want to play again.

```
The difference between what you rolled and your target number is:  -7
You lost! Sorry  :(
Do you want to play again? y/n
y
what is end:1
value of play:1
```

2.2.0

Same as 2.1.0 but fixing loops with either returning to false or true or if needed breaks. Still has the same output as 2.1.0.

2.3.0

Cleaned up the game by removing unnecessary sentences and adding cleanness to the output. Starting to develop and add different concepts and algorithms within the code.

2.4.0

Added bubble sort and selection sort within the program. Added prototypes per sorting algorithms and completed game.

```
Bubble sort dice rolls:
2 2 3 6 8 9 9 12 16 18 23
Selection sort dice rolls:
2 2 3 6 8 9 9 12 16 18 23
```

4 Pseudocode

intialize variables
calculate random time seed
display prototypes
intialize prototypes
display prototype
print prototypes
return

display intro prototype

display target and dice number prototype

display dice rolls prototype

display total/difference and win prototype

void swap prototype
find bubble prototype
find selection sort prototype
find linear sort prototype
printing array prototype
find size of the arry prototype
fill the array prototype
bool again prototype

5 Concepts Used

chapter ─ topic ─── (points) ─── where in line #

Chapter 2

- ☑ libraries (5) ───#10-16
- ☑ integers(1) ───#37
- ☑ characters(1)───#40
- ☑ strings(1)───#41
- ☐ floats(1)───
- ☑ bools(1)───#49
- ☑ comments 20% (2)───#38-43

Chapter 3

- ☐ type casting(1)───
- ☑ formatting output(1)───#70
- ☑ strings(1)───#41
- ☑ math library(1)───#52

Chapter 4

- ☑ if(1)───#95
- ☑ if-else(1)───#142
- ☑ nesting(1)───#125-131
- ☑ if-else if(1)───#131-142
- ☑ logical operator (1)───#104
- ☑ validating user input(1)───#95
- ☐ conditional operator(1)───
- ☑ switch(1)───#367

Chapter 5

- ☑ increment/decrement1(1)───#127
- ☑ while(1)───#91
- ☐ do while(1)───#
- ☑ for loop(1)───#391
- ☐ files input/output(2)───#

Chapter 6

- ☑ function prototype(4)───#347
- ☑ pass by value(4)───#381
- ☑ return(4)───#378
- ☑ returning boolean(4)───#447
- ☐ static variable(4)───#
- ☐ defaulted arguments(4)───#

- [x] pass by reference(4)——#381
- [ ] overloading(5)——#
- [x] exit() function(4)——#461

Chapter 7

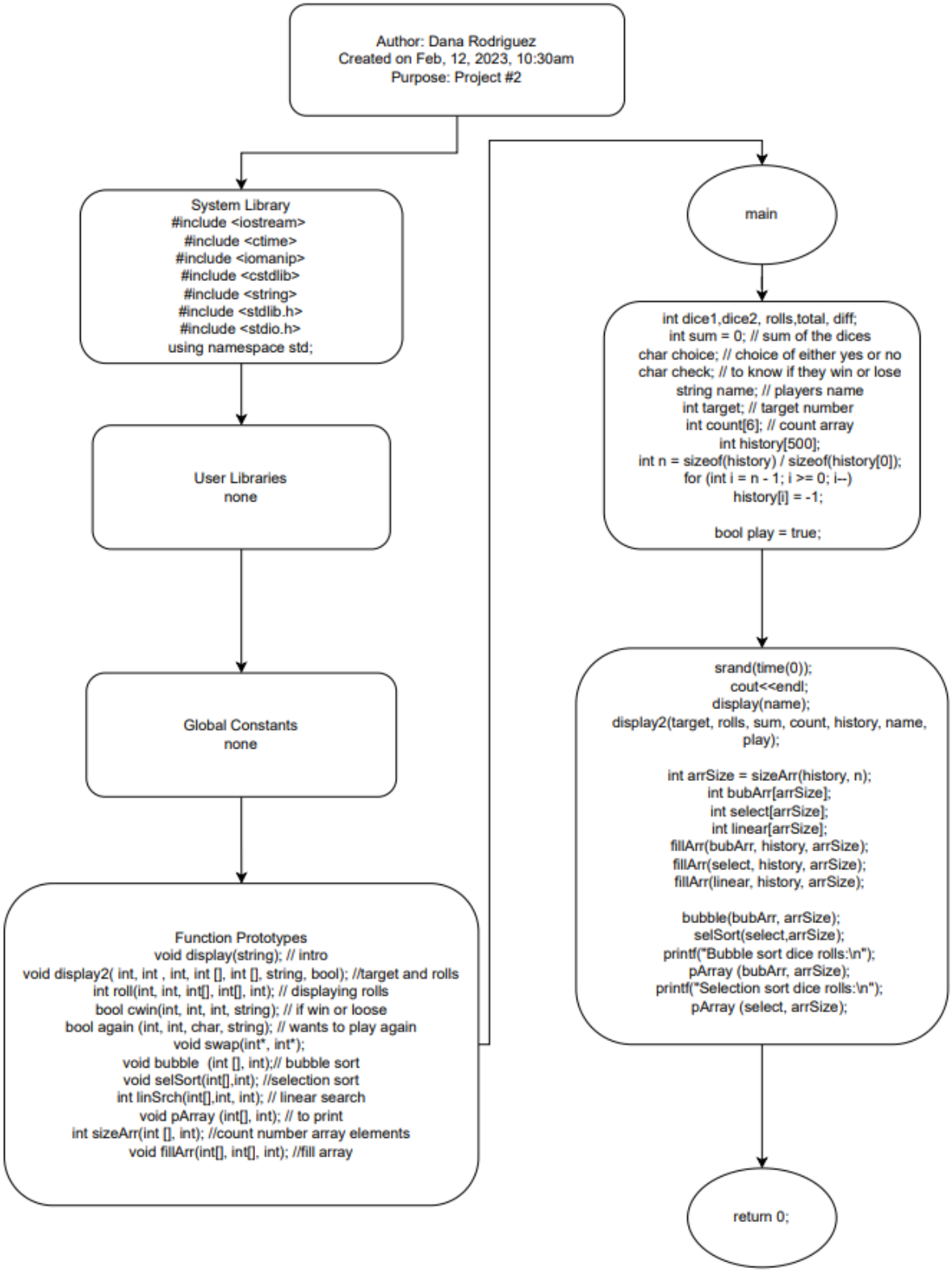
- [x] single dimensional arrays(3)——#43
- [x] parallel arrays(2)——#436
- [x] single dimension as function arguments(2)——#426
- [ ] 2 dimensional arrays(2)——#
- [ ] STL vectors(2)——#
- [x] passing arrays to and from functions(5)——#418
- [ ] passing vectors to and from functions(5)——#

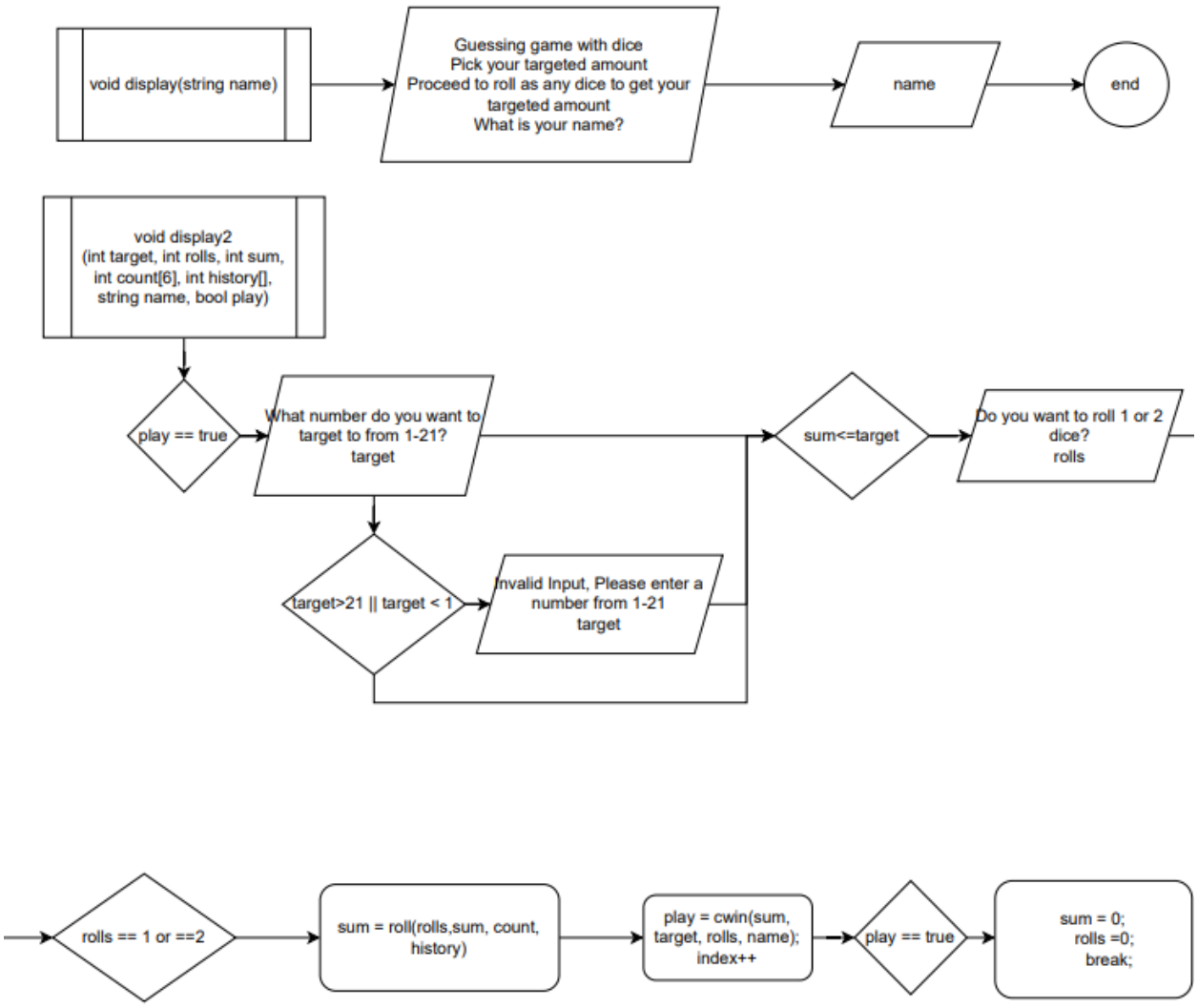
Chapter 8

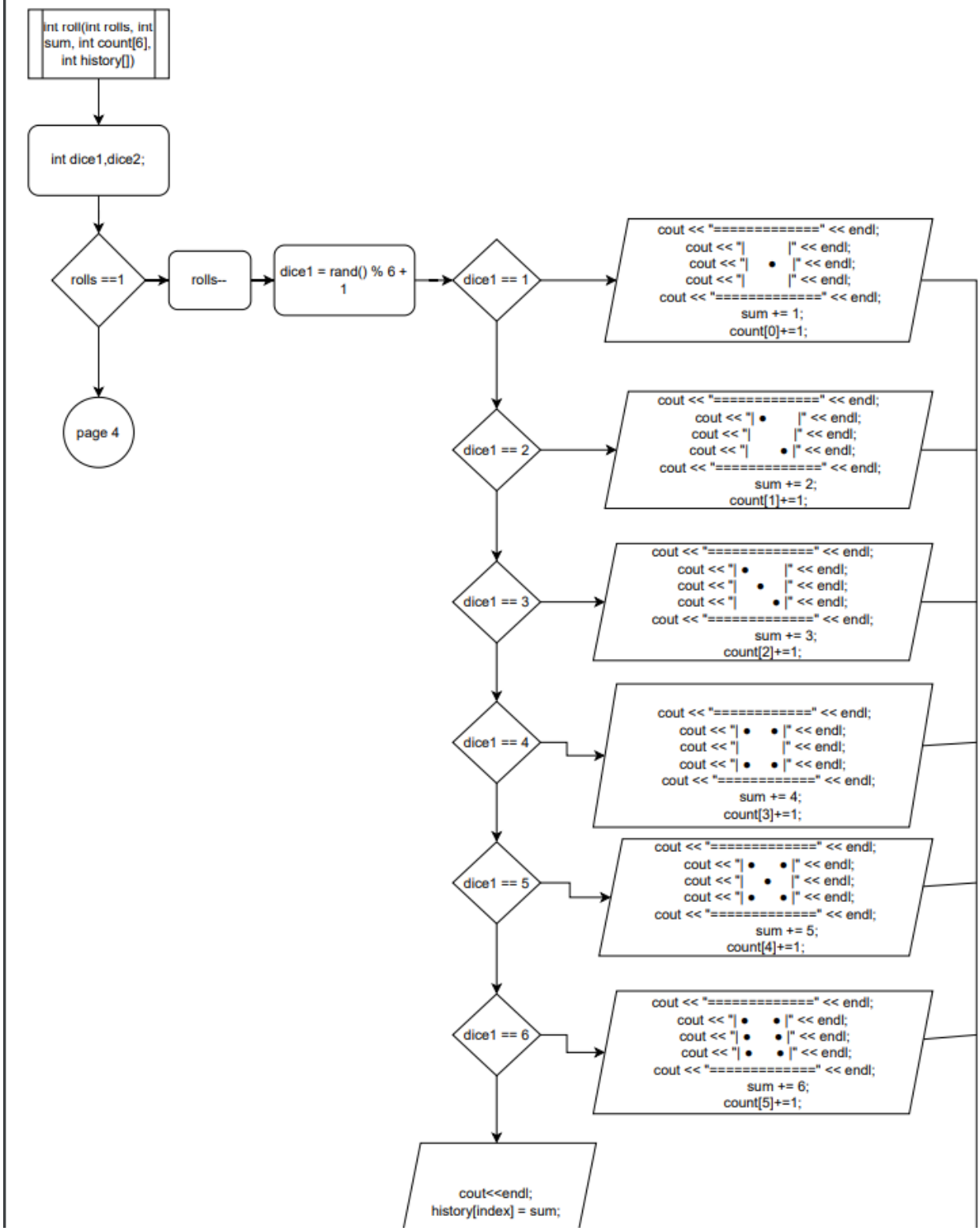
- [x] bubble sort(4)——#389
- [x] selection sort(4)——#399
- [x] linear or binary search(4)——#410

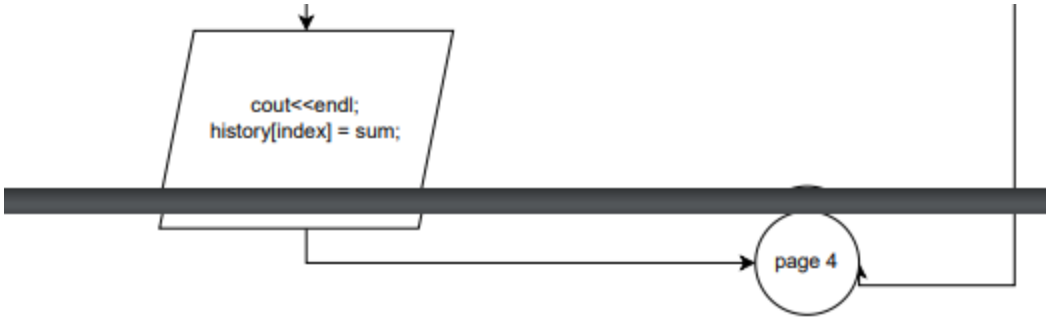
total:79

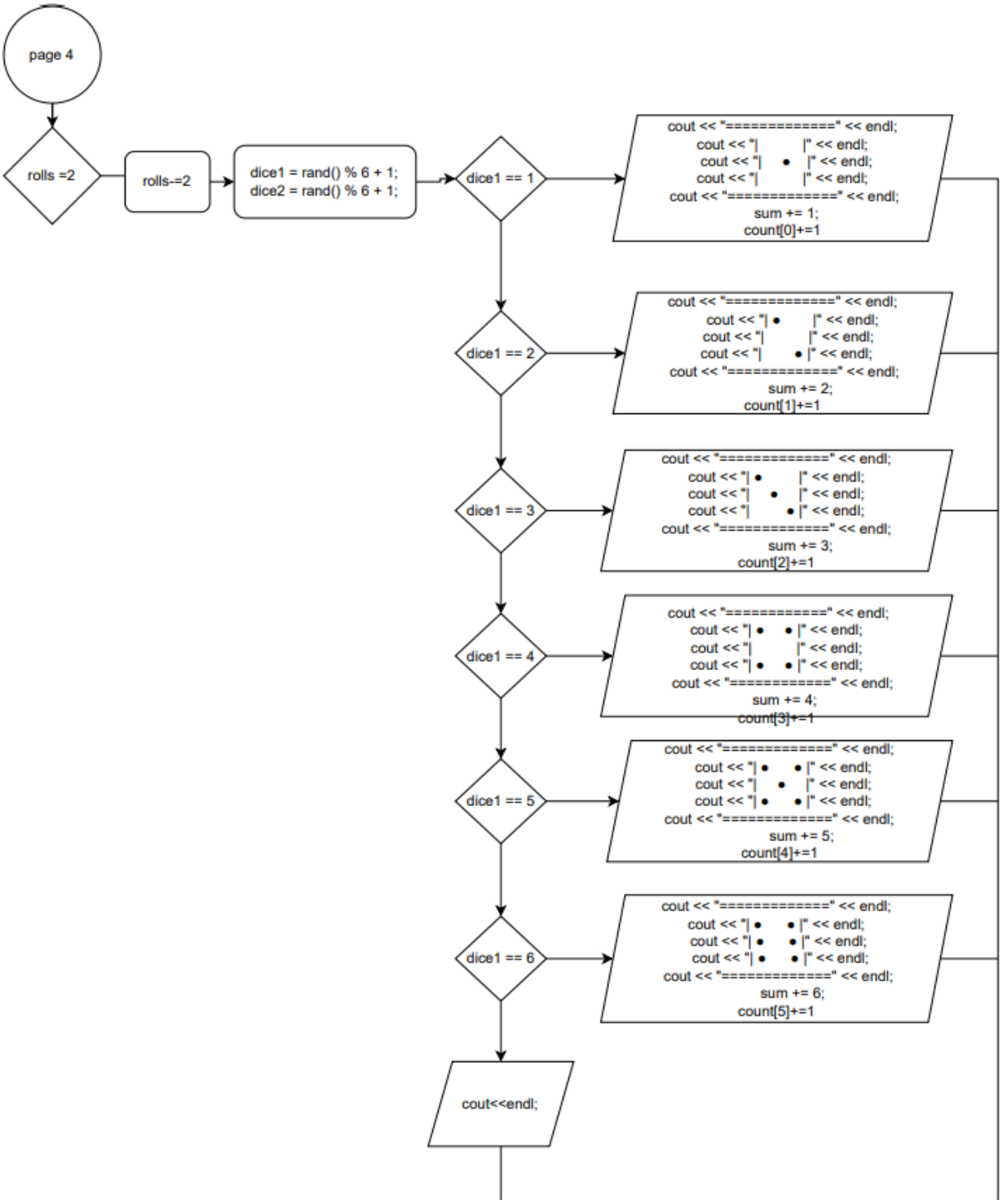
6 Flowchart

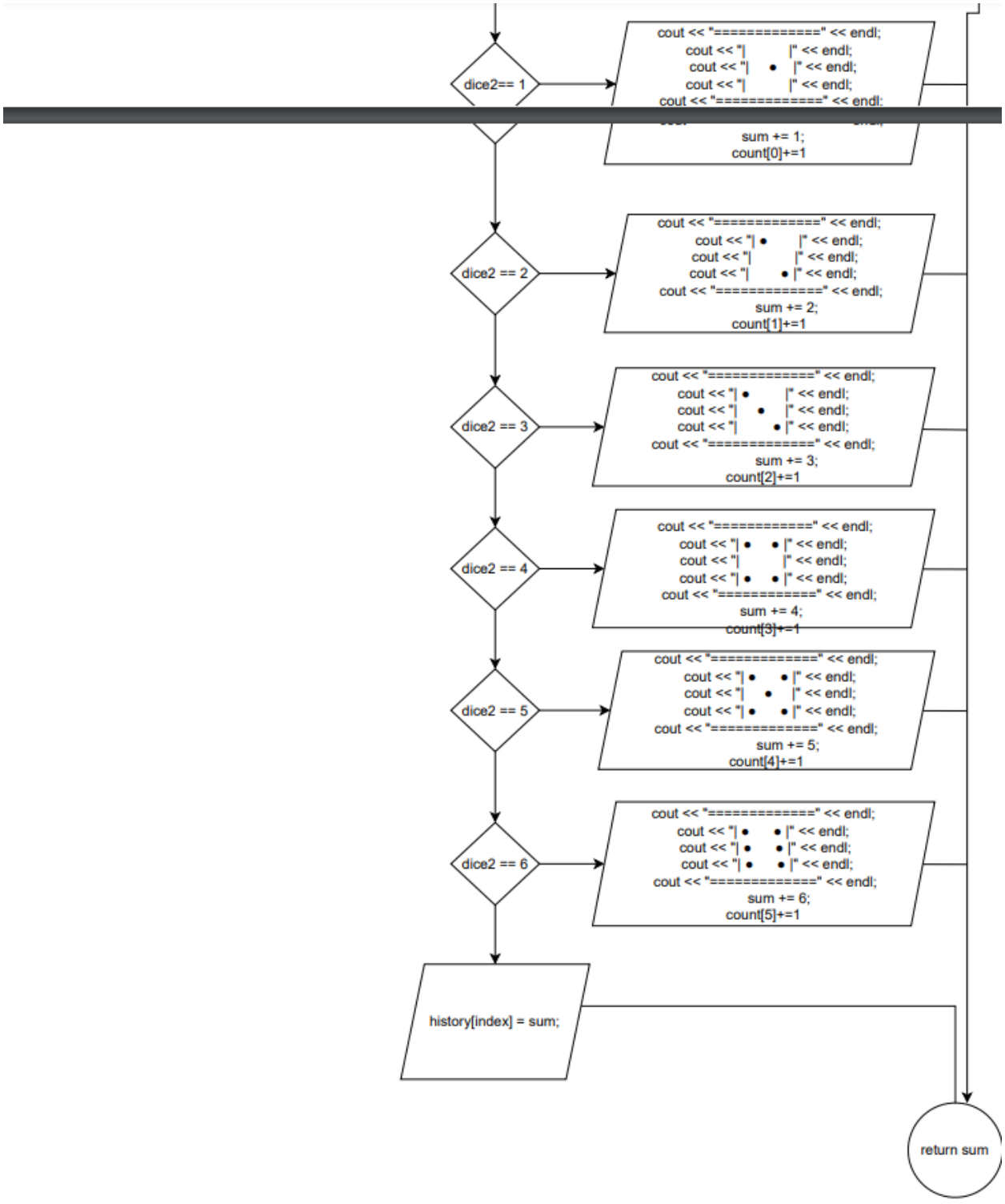

Author: Dana Rodriguez
Created on Feb, 12, 2023, 10:30am
Purpose: Project #2
System Library
#include <iostream>
#include <ctime>
#include <iomanip>
#include <cstdlib>
#include <string>
#include <stdlib.h>
#include <stdio.h>
using namespace std;
User Libraries
none
Global Constants
none
Function Prototypes
void display(string); // intro
void display2(int, int , int, int [], int [], string, bool); //target and rolls
int roll(int, int, int[], int[], int); // displaying rolls
bool cwin(int, int, int, string); // if win or loose
bool again (int, int, char, string); // wants to play again
void swap(int*, int*);
void bubble (int [], int);// bubble sort
void selSort(int[],int); //selection sort
int linSrch(int[],int, int); // linear search
void pArray (int[], int); // to print
int sizeArr(int [], int); //count number array elements
void fillArr(int[], int[], int); //fill array
main
int dice1,dice2, rolls,total, diff;
int sum = 0; // sum of the dices
char choice; // choice of either yes or no
char check; // to know if they win or lose
string name; // players name
int target; // target number
int count[6]; // count array
int history[500];
int n = sizeof(history) / sizeof(history[0]);
for (int i = n - 1; i >= 0; i--)
history[i] = -1;
bool play = true;
srand(time(0));
cout<<endl;
display(name);
display2(target, rolls, sum, count, history, name,
play);
int arrSize = sizeArr(history, n);
int bubArr[arrSize];
int select[arrSize];
int linear[arrSize];
fillArr(bubArr, history, arrSize);
fillArr(select, history, arrSize);
fillArr(linear, history, arrSize);
bubble(bubArr, arrSize);
selSort(select,arrSize);
printf("Bubble sort dice rolls:\n");
pArray (bubArr, arrSize);
printf("Selection sort dice rolls:\n");
pArray (select, arrSize);
return 0;

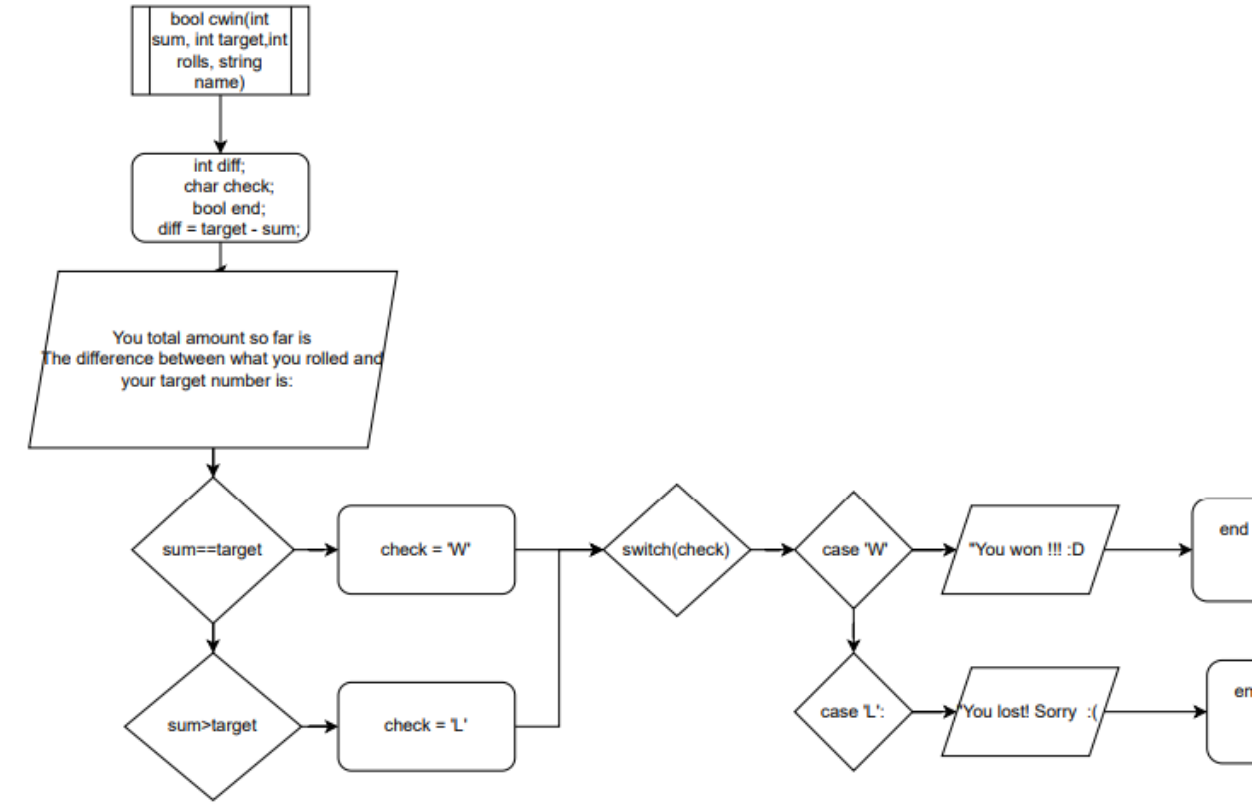

void display(string name)
Guessing game with dice
Pick your targeted amount
Proceed to roll as any dice to get your targeted amount
What is your name?
name
end
void display2
(int target, int rolls, int sum, int count[6], int history[], string name, bool play)
play == true
What number do you want to target to from 1-21?
target
sum<=target
Do you want to roll 1 or 2 dice?
rolls
target>21 || target < 1
Invalid Input, Please enter a number from 1-21
target
rolls == 1 or ==2
sum = roll(rolls,sum, count, history)
play = cwin(sum, target, rolls, name);
index++
play == true
sum = 0;
rolls =0;
break;


int roll(int rolls, int sum, int count[6], int history[])
int dice1,dice2;
rolls ==1
rolls--
dice1 = rand() % 6 + 1
page 4
dice1 == 1
cout << "=============" << endl;
cout << "| |" << endl;
cout << "| ● |" << endl;
cout << "| |" << endl;
cout << "=============" << endl;
sum += 1;
count[0]+=1;
dice1 == 2
cout << "=============" << endl;
cout << "| ● |" << endl;
cout << "| |" << endl;
cout << "| ● |" << endl;
cout << "=============" << endl;
sum += 2;
count[1]+=1;
dice1 == 3
cout << "=============" << endl;
cout << "| ● |" << endl;
cout << "| ● |" << endl;
cout << "| ● |" << endl;
cout << "=============" << endl;
sum += 3;
count[2]+=1;
dice1 == 4
cout << "=============" << endl;
cout << "| ● ● |" << endl;
cout << "| |" << endl;
cout << "| ● ● |" << endl;
cout << "=============" << endl;
sum += 4;
count[3]+=1;
dice1 == 5
cout << "=============" << endl;
cout << "| ● ● |" << endl;
cout << "| ● |" << endl;
cout << "| ● ● |" << endl;
cout << "=============" << endl;
sum += 5;
count[4]+=1;
dice1 == 6
cout << "=============" << endl;
cout << "| ● ● |" << endl;
cout << "| ● ● |" << endl;
cout << "| ● ● |" << endl;
cout << "=============" << endl;
sum += 6;
count[5]+=1;
cout<<endl;
history[index] = sum;


cout<<endl;
history[index] = sum;
page 4

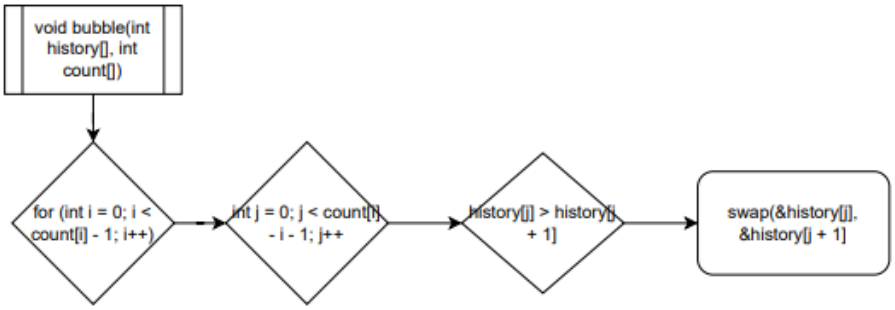

page 4
rolls =2
rolls-=2
dice1 = rand() % 6 + 1;
dice2 = rand() % 6 + 1;
dice1 == 1
cout << "==========" << endl;
cout << "| |" << endl;
cout << "| ● |" << endl;
cout << "| |" << endl;
cout << "==========" << endl;
sum += 1;
count[0]+=1
dice1 == 2
cout << "==========" << endl;
cout << "| ● |" << endl;
cout << "| |" << endl;
cout << "| ● |" << endl;
cout << "==========" << endl;
sum += 2;
count[1]+=1
dice1 == 3
cout << "==========" << endl;
cout << "| ● |" << endl;
cout << "| ● |" << endl;
cout << "| ● |" << endl;
cout << "==========" << endl;
sum += 3;
count[2]+=1
dice1 == 4
cout << "==========" << endl;
cout << "| ● ● |" << endl;
cout << "| |" << endl;
cout << "| ● ● |" << endl;
cout << "==========" << endl;
sum += 4;
count[3]+=1
dice1 == 5
cout << "==========" << endl;
cout << "| ● ● |" << endl;
cout << "| ● |" << endl;
cout << "| ● ● |" << endl;
cout << "==========" << endl;
sum += 5;
count[4]+=1
dice1 == 6
cout << "==========" << endl;
cout << "| ● ● |" << endl;
cout << "| ● ● |" << endl;
cout << "| ● ● |" << endl;
cout << "==========" << endl;
sum += 6;
count[5]+=1
cout<<endl;

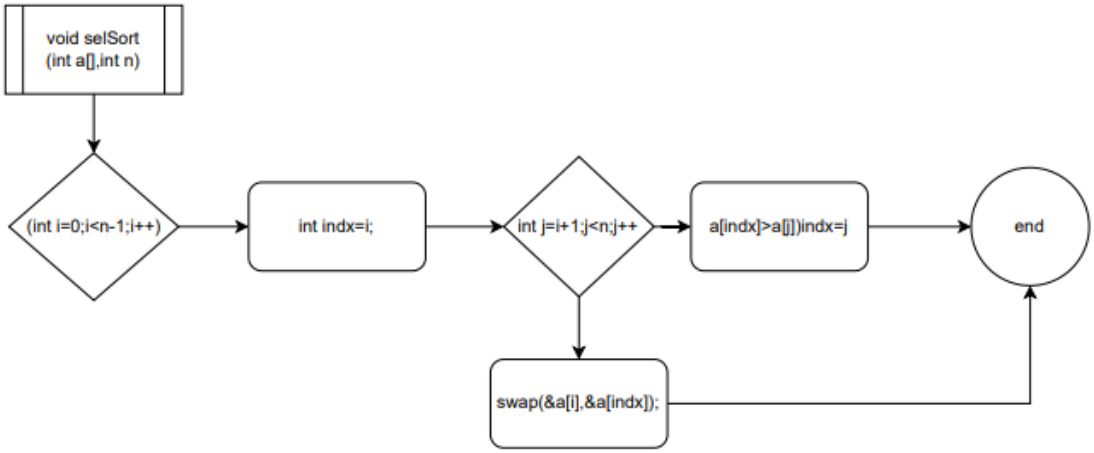

dice2== 1
cout << "==========" << endl;
cout << "| |" << endl;
cout << "| ● |" << endl;
cout << "| |" << endl;
cout << "==========" << endl;
sum += 1;
count[0]+=1
dice2 == 2
cout << "==========" << endl;
cout << "| ● |" << endl;
cout << "| |" << endl;
cout << "| ● |" << endl;
cout << "==========" << endl;
sum += 2;
count[1]+=1
dice2 == 3
cout << "==========" << endl;
cout << "| ● |" << endl;
cout << "| ● |" << endl;
cout << "| ● |" << endl;
cout << "==========" << endl;
sum += 3;
count[2]+=1
dice2 == 4
cout << "==========" << endl;
cout << "| ● ● |" << endl;
cout << "| |" << endl;
cout << "| ● ● |" << endl;
cout << "==========" << endl;
sum += 4;
count[3]+=1
dice2 == 5
cout << "==========" << endl;
cout << "| ● ● |" << endl;
cout << "| ● |" << endl;
cout << "| ● ● |" << endl;
cout << "==========" << endl;
sum += 5;
count[4]+=1
dice2 == 6
cout << "==========" << endl;
cout << "| ● ● |" << endl;
cout << "| ● ● |" << endl;
cout << "| ● ● |" << endl;
cout << "==========" << endl;
sum += 6;
count[5]+=1
history[index] = sum;
return sum

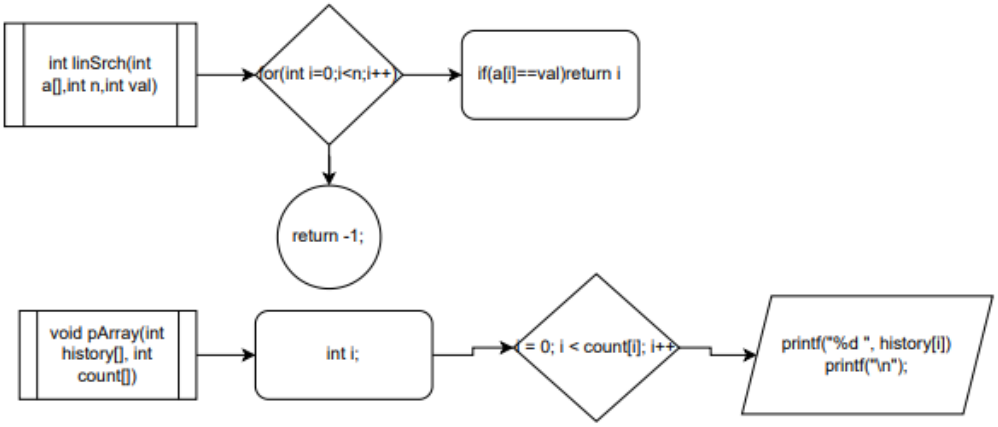

bool cwin(int sum, int target,int rolls, string name)
int diff;
char check;
bool end;
diff = target - sum;
You total amount so far is
The difference between what you rolled and
your target number is:
sum==target
check = 'W'
switch(check)
case 'W'
"You won !!! :D
end
sum>target
check = 'L'
case 'L':
"You lost! Sorry :(

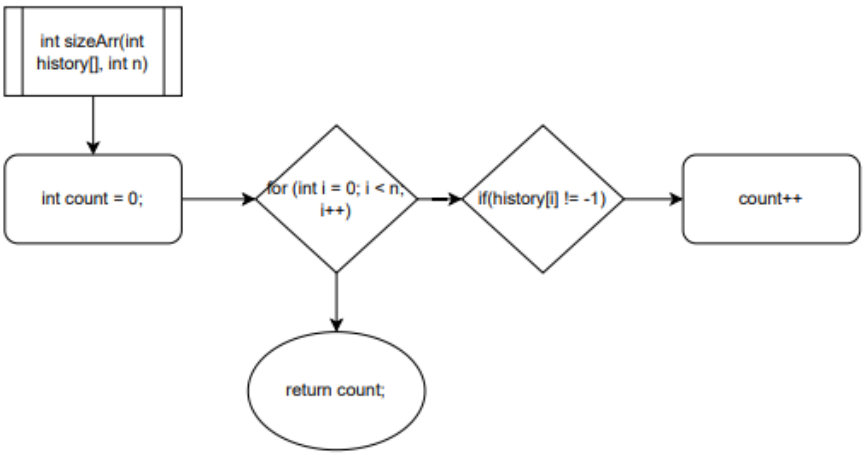

= again(sum, rolls,
check, name);
return end;
return end;
d = again(sum, rolls,
check, name);
return end;

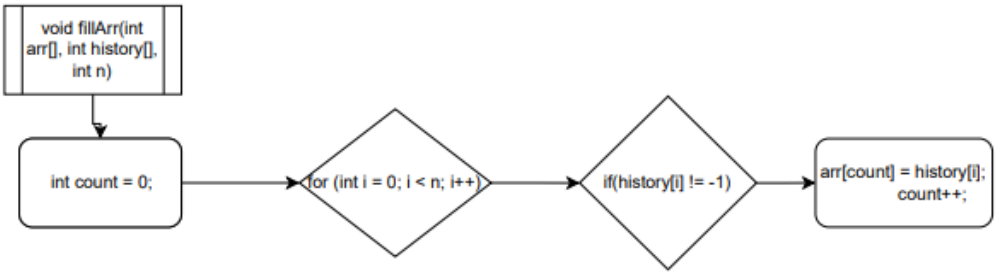

void swap(int* xp, int* yp)
int temp = *xp;
*xp = *yp;
*yp = temp;
end;

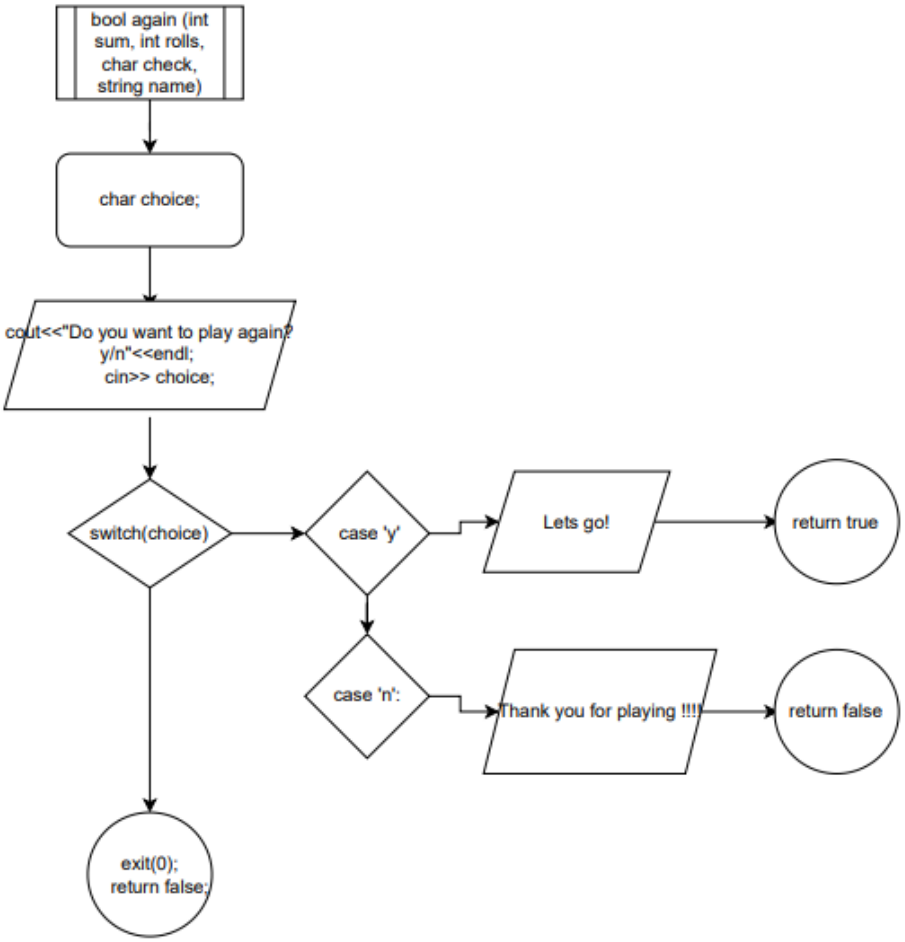

void bubble(int history[], int count[])
for (int i = 0; i < count[i] - 1; i++)
int j = 0; j < count[i] - i - 1; j++
history[j] > history[j + 1]
swap(&history[j], &history[j + 1]


void selSort (int a[],int n)
(int i=0;i<n-1;i++)
int indx=i;
int j=i+1;j<n;j++
a[indx]>a[j])indx=j
end
swap(&a[i],&a[indx]);


int linSrch(int a[],int n,int val)
for(int i=0;i<n;i++)
if(a[i]==val)return i
return -1;
void pArray(int history[], int count[])
int i;
= 0; i < count[i]; i++
printf("%d ", history[i])
printf("\n");


int sizeArr(int history[], int n)
int count = 0;
for (int i = 0; i < n; i++)
if(history[i] != -1)
count++
return count;


void fillArr(int arr[], int history[], int n)
int count = 0;
for (int i = 0; i < n; i++)
if(history[i] != -1)
arr[count] = history[i];
count++;


bool again (int sum, int rolls, char check, string name)
char choice;
cout<<"Do you want to play again? y/n"<<endl; cin>> choice;
switch(choice)
case 'y'
Lets go!
return true
case 'n':
Thank you for playing !!!!
return false
exit(0); return false;

7 Program Code

```
1
2   /*
3    * File:   main.cpp
4    * Author: danak
5    * Created on Feb 2,2023
6    * Purpose: Project #1
7    */
8   //System Lib
9
10  #include <iostream> //Input Output Library
11  #include <ctime>
12  #include <iomanip>
13  #include <cstdlib>
14  #include <string>
15  #include <stdlib.h>
16  #include <stdio.h>
17  using namespace std;
18
19  void display(string); // intro
20  void display2( int, int , int, int [], int [], string, bool); //target and rolls
21  int roll(int, int, int[], int[], int); // displaying rolls
22  bool cwin(int, int, int, string); // if win or loose
23  bool again (int, int, char, string); // wants to play again
24  void swap(int*, int*);
25  void bubble  (int [], int);// bubble sort
26  void selSort(int[],int); //selection sort
27  int linSrch(int[],int, int); // linear search
28  void pArray (int[], int); // to print
29  int sizeArr(int [], int); //count number array elements
30  void fillArr(int[], int[], int); //fill array
31
32
33  int main(int argc, char** argv) {
34
35      //variables
36
37      int dice1,dice2, rolls,total, diff;
38      int sum = 0; // sum of the dices
39      char choice; // choice of either yes or no
40      char check; // to know if they win or lose
41      string name; // players name
42      int target; // target number
43      int count[6]; // count array
44      int history[500];
45      int n = sizeof(history) / sizeof(history[0]);
46      for (int i = n - 1; i >= 0; i--)
47          history[i] = -1;
48
49      bool play = true;
50
```

d

```
78  void display(string name){
79      //introduction to the game
80      cout<< " \t Guessing game with dice\n";
81      cout<< "Pick your targeted amount"<<endl;
82      cout<< "Proceed to roll as any dice to get your targeted amount"<<endl;
83      // ask for players name
84      cout<< "What is your name?"<<endl;
85      cin>> name;
86  }
87
```

```
88  void display2(int target, int rolls, int sum, int count[6], int history[], string name, bool play){
89
90      int index = 0;
91      while(play == true){
92          cout<< "What number do you want to target to from 1-21?"<<endl;
93          cin >> target;
94          // input validation
95          if (target>21 || target < 1){
96              cout<<"Invalid Input, Please enter a number from 1-21"<<endl;
97              cin >> target;
98          }
99          //if sum of the dice is less than the target, continue
100         while (sum<=target){
101             cout<< "Do you want to roll 1 or 2 dice? "<<endl;
102             cin >> rolls;
103
104             if (rolls == 1 || rolls == 2)
105             {
106                 sum = roll(rolls,sum, count, history, index);
107             }
108             play = cwin(sum, target, rolls, name);
109             index++;
110
111             if (play == true)
112             {
113                 sum = 0;
114                 rolls =0;
115                 break;
116             }
117         }
118     }
119 }
120
```

```
121 int roll(int rolls, int sum, int count[6], int history[], int index){
122
123     int dice1,dice2;
124
125     if (rolls == 1){
126     //subtract one
127         rolls--;
128         //dice #1 and randomize the number
129         //value of dice #1 (1-6)
            dice1 = rand() % 6 + 1;
131         if (dice1 == 1)
132         {
133             cout << "=============" << endl;
134             cout << "|           |" << endl;
135             cout << "|     ●     |" << endl;
136             cout << "|           |" << endl;
137             cout << "=============" << endl;
138                 // adds to sum
139                 sum += 1;
140                 count[0]+=1;
141         }//if
142         else if (dice1 == 2)
143         {
144             cout << "=============" << endl;
145             cout << "| ●         |" << endl;
146             cout << "|           |" << endl;
147             cout << "|         ● |" << endl;
148             cout << "=============" << endl;
149                 // adds to sum
150                 sum += 2;
151                 count[1]+=1;
152         }//else if
153         else if (dice1 == 3)
154         {
155             cout << "=============" << endl;
156             cout << "| ●         |" << endl;
157             cout << "|     ●     |" << endl;
158             cout << "|         ● |" << endl;
159             cout << "=============" << endl;
160                 // adds to sum
161                 sum += 3;
162                 count[2]+=1;
163         }//else if
164         else if (dice1 == 4)
```

```
165         {
166             cout << "============" << endl;
167             cout << "| ●      ● |" << endl;
168             cout << "|           |" << endl;
169             cout << "| ●      ● |" << endl;
170             cout << "============" << endl;
171                 // adds to sum
172                 sum += 4;
173                 count[3]+=1;
174         }//else if
175         else if (dice1 == 5)
176         {
177             cout << "=============" << endl;
178             cout << "| ●       ● |" << endl;
179             cout << "|     ●     |" << endl;
180             cout << "| ●       ● |" << endl;
181             cout << "=============" << endl;
182                 // adds to sum
183                 sum += 5;
184                 count[4]+=1;
185         }//else if
186         else if (dice1 == 6)
187         {
188             cout << "=============" << endl;
189             cout << "| ●       ● |" << endl;
190             cout << "| ●       ● |" << endl;
191             cout << "| ●       ● |" << endl;
192             cout << "=============" << endl;
193                 // adds to sum
194                 sum += 6;
195                 count[5]+=1;
196         }//else if
197         cout << endl;
198         history[index] = sum;
199     }
200         //dice #2
```

```
201     else if (rolls == 2){
202
203         rolls -= 2;
204         //value of dice #1 (1-6)
205         dice1 = rand() % 6 + 1;
206         //value of dice #2 (1-6)
207         dice2 = rand() % 6 + 1;
208         if (dice1 == 1)
209         {
210             cout << "=============" << endl;
211             cout << "|           |" << endl;
212             cout << "|     ●     |" << endl;
213             cout << "|           |" << endl;
214             cout << "=============" << endl;
215                 // adds to sum
216                 sum += 1;
217                 count[0]+=1;
218         }//if
219         else if (dice1 == 2)
220         {
221             cout << "=============" << endl;
222             cout << "| ●         |" << endl;
223             cout << "|           |" << endl;
224             cout << "|         ● |" << endl;
225             cout << "=============" << endl;
226                 // adds to sum
227                 sum += 2;
228                 count[1]+=1;
229         }//else if
230         else if (dice1 == 3)
231         {
232             cout << "=============" << endl;
233             cout << "| ●         |" << endl;
234             cout << "|     ●     |" << endl;
235             cout << "|         ● |" << endl;
236             cout << "=============" << endl;
237                 // adds to sum
238                 sum += 3;
239                 count[2]+=1;
240         }//else if
241         else if (dice1 == 4)
242         {
243             cout << "============" << endl;
244             cout << "| ●     ● |" << endl;
245             cout << "|         |" << endl;
246             cout << "| ●     ● |" << endl;
247             cout << "============" << endl;
```

```
248                 // adds to sum
249                 sum += 4;
250                 count[3]+=1;
251             }//else if
252             else if (dice1 == 5)
253             {
254                 cout << "=============" << endl;
255                 cout << "| ●       ● |" << endl;
256                 cout << "|     ●     |" << endl;
257                 cout << "| ●       ● |" << endl;
258                 cout << "=============" << endl;
259                 // adds to sum
260                 sum += 5;
261                 count[4]+=1;
262             }//else if
263             else if (dice1 == 6)
264             {
265                 cout << "=============" << endl;
266                 cout << "| ●       ● |" << endl;
267                 cout << "| ●       ● |" << endl;
268                 cout << "| ●       ● |" << endl;
269                 cout << "=============" << endl;
270                 // adds to sum
271                 sum += 6;
272                 count[5]+=1;
273             }//else if
274
275             cout << endl;
276
277             if (dice2 == 1)
278             {
279                 cout << "=============" << endl;
280                 cout << "|           |" << endl;
281                 cout << "|     ●     |" << endl;
282                 cout << "|           |" << endl;
283                 cout << "=============" << endl;
284                 // adds to sum
285                 sum += 1;
286                 count[0]+=1;
287             }//if
288             else if (dice2 == 2)//
289             {
290                 cout << "=============" << endl;
291                 cout << "| ●         |" << endl;
292                 cout << "|           |" << endl;
293                 cout << "|         ● |" << endl;
294                 cout << "=============" << endl;
295                 // adds to sum
296                 sum += 2;
```

```
297                 count[1]+=1;
298             }//else if
299             else if (dice2 == 3)
300             {
301                 cout << "=============" << endl;
302                 cout << "| ●         |" << endl;
303                 cout << "|     ●     |" << endl;
304                 cout << "|         ● |" << endl;
305                 cout << "=============" << endl;
306                     // adds to sum
307                     sum += 3;
308                     count[2]+=1;
309             }//else if
310             else if (dice2 == 4)
311             {
312                 cout << "=============" << endl;
313                 cout << "| ●     ● |" << endl;
314                 cout << "|           |" << endl;
315                 cout << "| ●     ● |" << endl;
316                 cout << "=============" << endl;
317                     // adds to sum
318                     sum += 4;
319                     count[3]+=1;
320             }//else if
321             else if (dice2 == 5)
322             {
323                 cout << "=============" << endl;
324                 cout << "| ●     ● |" << endl;
325                 cout << "|     ●     |" << endl;
326                 cout << "| ●     ● |" << endl;
327                 cout << "=============" << endl;
328                     // adds to sum
329                     sum += 5;
330                     count[4]+=1;
331             }//else if
332             else if (dice2 == 6)
333             {
334                 cout << "=============" << endl;
335                 cout << "| ●     ● |" << endl;
336                 cout << "| ●     ● |" << endl;
337                 cout << "| ●     ● |" << endl;
338                 cout << "=============" << endl;
339                     // adds to sum
340                     sum += 6;
```

```
341                 count[5]+=1;
342             }
343             history[index] = sum;//else if
344         }
345         return sum;
346     }
347 bool cwin(int sum, int target,int rolls, string name){
348     int diff;
349     char check;
350     bool end;
351     diff = target - sum;
352 
353     // states total amount
354     cout <<"You total amount so far is " <<sum<<endl;
355     cout << "The difference between what you rolled and your target number is:  "<<diff<<endl;
356     //to find out what they need to go to left
357 
358     //if the sum is the target go to 'w'
359     if(sum==target){
360         check = 'W';
361     }
362     //if the sum is greater than the target, go to 'l'
363     else if (sum>target){
364         check= 'L';
365     }
366     //the message to the player of what their results are
367     switch(check){
368         case 'W':
369             cout << "You won !!! :D "<<endl;
370             end = again(sum, rolls, check, name);
371             return end;
372 
373         case 'L':
374             cout << "You lost! Sorry  :( "<<endl;
375             end = again(sum, rolls, check, name);
376             return end;
377     }
378     return end;
379 }
380 //
```

```
381 void swap(int* xp, int* yp){
382     int temp = *xp;
383 
384     *xp = *yp;
385     *yp = temp;
386 }
387 
388 // bubble sorting
389 void bubble(int history[], int n){
390     int i, j;
391     for (int i = 0; i < n; i++)
392         // Last i elements are already in place
393         for ( int j = 0; j < n - i - 1; j++)
394             if (history[j] > history[j + 1])
395                 swap(&history[j], &history[j + 1]);
396 }
397 
398 // selection sort
399 void selSort(int a[],int n){
400     for(int i=0;i<n-1;i++){
401         int indx=i;
402         for(int j=i+1;j<n;j++){
403             if(a[indx]>a[j])indx=j;
404         }
405         swap(&a[i],&a[indx]);
406     }
407 }
408 
409 // linear search
410 int linSrch(int a[],int n,int val){
411     for(int i=0;i<n;i++){
412         if(a[i]==val)return i;
413     }
414     return -1;
415 }
416 
```

```
417     // Function to print an array
418     void pArray(int arr[], int n){
419         int i;
420
421         for (i = 0; i < n; i++)
422             printf("%d ", arr[i]);
423         printf("\n");
424     }
425     // figuring out the size within an array
426     int sizeArr(int history[], int n){
427         int count = 0;
428         for (int i = 0; i < n; i++){
429             if(history[i] != -1){
430                 count++;
431             }
432         }
433         return count;
434     }
435     // filling the array of numbers
436     void fillArr(int arr[], int history[], int n){
437         int count = 0;
438         for (int i = 0; i < n; i++){
439             if(history[i] != -1){
440                 arr[count] = history[i];
441                 count++;
442             }
443         }
444     }
445
446     //asking if they want to play again
447     bool again (int sum, int rolls, char check, string name){
448         char choice;
449         cout<<"Do you want to play again? y/n"<<endl;
450         cin>> choice;
451         switch(choice){
452             // go back to loop
453             case 'y': case 'Y':
454                 cout<<"Lets go!"<<endl;
455                 return true;
456             //ends game
457             case 'n' : case 'N':
458                 cout<< "Thank you for playing !!!!"<<endl;
459                 return false;
460         }
461         exit(0);
462         return false;
463     }
```