|  |  |
| --- | --- |
| **Application/ Program name:** | Die or Lab 1-1 |
| **Written by:** | Bailey Nichols |

|  |
| --- |
| **Purpose or problem definition:** |
| The purpose of the program is to show what numbers are produced from the rolling of computerized dice. From the prompt given in the Lab 1 Mockups document “Your program will prompt the user to enter the number of dies in a set (4 through 6) that will be rolled together. The sum of the faces on the set will be the index to an array that holds the number of times this sum has occurred. You must also prompt your user for the number of times the set will be rolled. (2500, 3000, or 5000 times).” |
|  |
| **Program Procedures:** |
| * Your program will prompt the user to enter the number of dies in a set (4 through 6) that will be rolled together. * The sum of the faces on the set will be the index to an array that holds the number of times this sum has occurred. * You must also prompt your user for the number of times the set will be rolled. (2500, 3000, or 5000 times) * Once the rolls are completed the program will display a bar graph, like the one shown in notes * Bar graph will show the sum value, the number of times it was rolled and a bar of astrisks each one representing 2% of total (50x \* = 100%=) |
|  |
| **Algorithm/Processing/Conditions:** |
| **Inputs:** |
| The user will input to the console the amount of dice to be used  The user will input the number of times to roll the dice |
| **Processes:** |
| The program will get input from the user to decide how many dice and how many rolls and then will use these numbers to output a chart (as shown in notes) to the user. |
| **Outputs:** |
| Said chart shown in notes. |
|  |
| **Notes & Restriction:** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | ***Object*** | ***Members*** | ***Function*** | ***Exposure*** | ***Algorithm*** | | Die | Num | Roll dice, Display menu, | Roll, getNum |  |   ***The sum of the faces on the set will be the index to an array that holds the number of times this sum has occurred.*** |
|  |
| **Comments:** |
| Easy Peasy  (from chat helping other student)  [9:16 PM] Bailey:it seems complex but all you are doing is adding one to the location that is the value of the sum of the output each time you get an output.  [9:18 PM] Bailey: so each time i get a 36 from all the dice it adds one to the location 36 in the array, you print the array to make the graph and the number of times the (all six) dice rolled 36 in 5000 rolls, will be in the location array[36]  chart output  booboo@forest:~/Documents/Code/CIS023/Lab1/L1-1$ ./dieProgram  Enter the number of dies to use (4, 5, or 6)  or press enter to default to 6  Enter the number of times the 6  dice are to be rolled (2500, 3000 or 5000)  or press enter to default to 5000  For 6 dice, rolled 5000 times  Count %:--- 0 10 20 30 40 50 60 70 80 90 100  Sum count |....|....|....|....|....|....|....|....|....|....|  0 ( 0)  1 ( 0)  2 ( 0)  3 ( 0)  4 ( 0)  5 ( 0)  6 ( 0)  7 ( 0)  8 ( 2)  9 ( 9)  10 ( 27) \*\*  11 ( 28) \*\*  12 ( 47) \*\*\*\*  13 ( 67) \*\*\*\*\*\*  14 ( 139) \*\*\*\*\*\*\*\*\*\*\*\*\*  15 ( 190) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  16 ( 247) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  17 ( 308) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  18 ( 408) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  19 ( 404) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  20 ( 465) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  21 ( 437) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  22 ( 450) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  23 ( 406) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  24 ( 368) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  25 ( 284) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  26 ( 220) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  27 ( 188) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  28 ( 124) \*\*\*\*\*\*\*\*\*\*\*\*  29 ( 85) \*\*\*\*\*\*\*\*  30 ( 55) \*\*\*\*\*  31 ( 22) \*\*  32 ( 14) \*  33 ( 4)  34 ( 2)  35 ( 0)  36 ( 0) |