

## Program outputs:

### 1. Horspool string matching

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> cd "c:\U
_pool_string_match_2.cpp -o horse_pool_string_match_2 } ; if ($?) { .\horse_pool_str
enter the text: john_has_eaten_apple
enter the pattern to match: has
pattern match found at string index: 5
pattern match found at absolute position: 6
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> █
```

### 2. Heapsort

```
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> cd "
sort.cpp -o heap_sort } ; if ($?) { .\heap_sort }
enter the number of elements: 7
enter the elements: 4 5 2 6 7 1 3
current heap: 7 6 3 4 5 1 2
current heap: 6 5 3 4 2 1 7
current heap: 5 4 3 1 2 6 7
current heap: 4 2 3 1 5 6 7
current heap: 3 2 1 4 5 6 7
current heap: 1 2 3 4 5 6 7
heapsort done
current heap: 1 2 3 4 5 6 7
```

### 3. Binomial coefficient

```
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> cd "c
ial_coeff.cpp -o binomial_coeff } ; if ($?) { .\binomial_coeff }
Enter value of n: 5
Enter value of k: 2
Value of 5C2 is: 10
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> █
```

### 4. Knapsack

```
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> cd "c:\Users\adith\Doc
ack.cpp -o knapsack } ; if ($?) { .\knapsack }
enter no. of items: 4
enter the weights of the items: 2 2 3 6
enter the profits of the items: 12 16 11 15
enter the value of c: 11
0 0 0 0 0 0 0 0 0 0 0
0 0 12 12 12 12 12 12 12 12 12
0 0 16 16 28 28 28 28 28 28 28
0 0 16 16 28 28 28 39 39 39 39
0 0 16 16 28 28 28 39 39 39 43
max possible value: 43
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> █
```

## 5. Floyd-warshalls algorithm

```
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> cd "c:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not" && gcc _warshall.cpp -o floyd_warshall } ; if ($?) { .\floyd_warshall }
Enter number of vertices and edges: 4 5
Enter edge information (source, destination, weight):
0 1 5
0 2 10
1 2 3
2 1 2
2 3 6
Shortest distances between all pairs of vertices:
0 5 8 14
INF 0 3 9
INF 2 0 6
INF INF INF 0
PS C:\Users\adith\Documents\bmsce\2nd year\4th sem\DAA\DAA_pratice_not git> █
```