Jessica Chen
Collaborators: Andrew Jones, Sujaan Joshi

CSCI 303: Algorithms, HW 11

Due: 3:00 pm, Wednesday, 12/7

1. Problem 1

   (a) next fit

      i. Bin 1: $\frac{18}{34}$

      ii. Bin 1: $\frac{18}{34}, \frac{3}{34}$

      iii. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}$

      iv. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$

      v. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$
      Bin 3: $\frac{22}{34}$

      vi. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$
      Bin 3: $\frac{22}{34}, \frac{5}{34}$

      vii. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$
      Bin 3: $\frac{22}{34}, \frac{5}{34}$
      Bin 4: $\frac{12}{34}$

      viii. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$
      Bin 3: $\frac{22}{34}, \frac{5}{34}$
      Bin 4: $\frac{12}{34}, \frac{8}{34}$

      ix. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$
      Bin 3: $\frac{22}{34}, \frac{5}{34}$
      Bin 4: $\frac{12}{34}, \frac{8}{34}$
      Bin 5: $\frac{20}{34}$

      x. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$
      Bin 3: $\frac{22}{34}, \frac{5}{34}$
      Bin 4: $\frac{12}{34}, \frac{8}{34}$
      Bin 5: $\frac{20}{34}, \frac{9}{34}$

      xi. Bin 1: $\frac{18}{34}, \frac{3}{34}$
      Bin 2: $\frac{16}{34}, \frac{4}{34}$

Bin 3: $\frac{22}{34}, \frac{5}{34}$
Bin 4: $\frac{12}{34}, \frac{8}{34}$
Bin 5: $\frac{20}{34}, \frac{9}{34}$ Bin 6: $\frac{14}{34}$

xii. Bin 1: $\frac{18}{34}, \frac{3}{34}$
Bin 2: $\frac{16}{34}, \frac{4}{34}$
Bin 3: $\frac{22}{34}, \frac{5}{34}$
Bin 4: $\frac{12}{34}, \frac{8}{34}$
Bin 5: $\frac{20}{34}, \frac{9}{34}$ Bin 6: $\frac{14}{34}, \frac{2}{34}$

(b) first fit

i. Bin 1: $\frac{18}{35}$

ii. Bin 1: $\frac{18}{35}, \frac{3}{34}$

iii. Bin 1: $\frac{18}{35}, \frac{3}{34}$
Bin 2: $\frac{16}{34}$

iv. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}$
Bin 2: $\frac{16}{34}$

v. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}$
Bin 2: $\frac{16}{34}$
Bin 3: $\frac{22}{34}$

vi. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}$
Bin 3: $\frac{22}{34}$

vii. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{12}{34}$
Bin 3: $\frac{22}{34}$

viii. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{12}{34}$
Bin 3: $\frac{22}{34}, \frac{8}{34}$

ix. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{12}{34}$
Bin 3: $\frac{22}{34}, \frac{8}{34}$
Bin 4: $\frac{20}{34}$

x. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{12}{34}$
Bin 3: $\frac{22}{34}, \frac{8}{34}$
Bin 4: $\frac{20}{34}, \frac{9}{34}$

xi. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{12}{34}$
Bin 3: $\frac{22}{34}, \frac{8}{34}$
Bin 4: $\frac{20}{34}, \frac{9}{34}$
Bin 5: $\frac{14}{34}$

xii. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}, \frac{2}{34}$
Bin 2: $\frac{16}{34}, \frac{12}{34}$
Bin 3: $\frac{22}{34}, \frac{8}{34}$
Bin 4: $\frac{20}{34}, \frac{9}{34}$
Bin 5: $\frac{14}{34}$

(c) best fit

i. Bin 1: $\frac{18}{35}$

ii. Bin 1: $\frac{18}{35}, \frac{3}{34}$

iii. Bin 1: $\frac{18}{35}, \frac{3}{34}$
Bin 2: $\frac{16}{34}$

iv. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}$
Bin 2: $\frac{16}{34}$
Bin 3: $\frac{22}{34}$

v. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}$
Bin 3: $\frac{22}{34}$

vi. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}$
Bin 3: $\frac{22}{34}, \frac{12}{34}$

vii. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{8}{34}$
Bin 3: $\frac{22}{34}, \frac{12}{34}$

viii. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{8}{34}$
Bin 3: $\frac{22}{34}, \frac{12}{34}$
Bin 4: $\frac{20}{34}$

ix. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{8}{34}, \frac{9}{34}$
Bin 3: $\frac{22}{34}, \frac{12}{34}$
Bin 4: $\frac{20}{34}$

x. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}$
Bin 2: $\frac{16}{34}, \frac{8}{34}, \frac{9}{34}$

Bin 3: $\frac{22}{34}, \frac{12}{34}$
Bin 4: $\frac{20}{34}, \frac{14}{34}$

xi. Bin 1: $\frac{18}{35}, \frac{3}{34}, \frac{4}{34}, \frac{5}{34}, \frac{2}{34}$
Bin 2: $\frac{16}{34}, \frac{8}{34}, \frac{9}{34}$
Bin 3: $\frac{22}{34}, \frac{12}{34}$
Bin 4: $\frac{20}{34}, \frac{14}{34}$

(d) first fit decreasing

i. Bin 1: $\frac{22}{35}$
ii. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}$
iii. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}$
Bin 3: $\frac{18}{34}$
iv. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
v. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
vi. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
vii. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}$
viii. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}$
ix. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}$
x. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}, \frac{4}{34}$
xi. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}, \frac{4}{34}, \frac{3}{34}$

xii. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}, \frac{4}{34}, \frac{3}{34}, \frac{2}{34}$

(e) best fit decreasing

i. Bin 1: $\frac{22}{35}$

ii. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}$

iii. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}$
Bin 3: $\frac{18}{34}$

iv. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$

v. Bin 1: $\frac{22}{35}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$

vi. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$

vii. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}$

viii. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}$

ix. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}$

x. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}, \frac{4}{34}$

xi. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}, \frac{4}{34}, \frac{3}{34}$

xii. Bin 1: $\frac{22}{35}, \frac{12}{34}$
Bin 2: $\frac{20}{34}, \frac{14}{34}$
Bin 3: $\frac{18}{34}, \frac{16}{34}$
Bin 4: $\frac{9}{34}, \frac{8}{34}, \frac{5}{34}, \frac{4}{34}, \frac{3}{34}, \frac{2}{34}$

The optimal solution is most likely the off-line best fit increasing algorithm. It minimizes the bins to the optimal solution of $M = 4$ bins overall.

According to theorem 10.2 the optimal number of bins required to pack $I = 12$ items is never more than $2M = 8$ and the optimal solution for next fit was 6 which was within the bounds.

According to theorem 10.3 for first fit the optimal number of bins required to pack $I = 12$ items is never more than $\lceil \frac{17}{10}M \rceil = 7$ and the optimal solution for first fit was 5 which was within the bounds.

According to Weiss for best fit the optimal number of bins required to pack $I = 12$ items is approximately never more than 1.7 times as bad as the optimal $\lceil \frac{17}{10}M \rceil = 7$ and the optimal solution for first fit was 4 which was within the bounds.

According to theorem 10.4 for first fit decreasing the optimal number of bins required to pack $I = 12$ items is never more than $\frac{4M+1}{3} = 6$ and the optimal solution for first fit was 4 which was within the bounds.

Best fit decreasing has similar results to first fit, because as you pack the biggest items first it becomes easier and easier to minimize lost bin space by fitting the largest items possible together.

2. Problem 2

(a)
```
for i <- 1 to m
        min_num(i) = 0
for j from 1 to n
        if stamp[j] < m
                min_num(stamp[j]) += 1
        elif stamp[j] == m
                return 1
        else
                break

for i <- 1 to m
        if min_num(i) = 0
                min <- infinity
                for j <- 1 to n
                        if stamps[j] < i
                                if min > min_num(i - stamps[j])
                                        min <- min_num(i - stamps[j])
                min_num(i) <- 1 + min
return min_num(m)
```

(b) The time complexity is $\Theta(mn)$

(c) To augment the algorithm to keep track of the minimum number of stamps of each denomination an array could be added storing all the minimum values. The array would be of size m (saved_ stamps). After generating the array, the generation of min_ num(i) could be replaced with a call to the appropriate spot in saved_ stamps.

```
for i <- 1 to m
        for j <- 1 to n
                saved_stamps[i][j] = 0
for i <- 1 to m
```

```
        min <- infinity
        for j <- 1 to n
                if stamps[j] < i
                        if min > saved_stamps(i - stamps[j])
                                min <- saved_stamps[i - stamps[j]][j]
                if stamps[j] == i
                        min <- 0
                        saved_stamps[i][j] <- 0
        saved_stamps[i][j] <- 1 + min
return saved_stamps

stamps_to_be_used_for(m)
        for j <- 1 to n
                if(saved_stamps[m][j] > 0)
                        print saved_stamps[m][j]
        return sum
```

(d) The augmentation increases space complexity of the algorithm originally given in 2a. The changes do not affect the overall time complexity reported in 2b, it only affects the constant that the time complexity is multiplied by, by making the memory accesses more predictable and because more operations are done to store the coin values.

2a Algorithm: NOTE that for each min_ num you must recur and recalculate min_ num each time. Resulting in a lot of recurrences/replications. min_ num (1) = 1s
min_ num (2) = 1
min_ num (4) = 1
min_ num (17) = 1
min_ num (21) = 1
min_ num (1) = 1
min_ num (2) = 1
min_ num (3) = min(min_ num (1), min_ num (2)) + 1= 2
min_ num (4) = 1
min_ num (5) = min(min_ num (1), min_ num (3), min_ num(4)) + 1= 2
min_ num (6) = min(min_ num (5), min_ num (4), min_ num(2)) + 1= 2
min_ num (7) = min(min_ num (6), min_ num (5), min_ num(3)) + 1= 3
min_ num (8) = min(min_ num (7), min_ num (6), min_ num(4)) + 1= 2
min_ num (9) = min(min_ num (8), min_ num (7), min_ num(5)) + 1= 3
min_ num (10) = min(min_ num (9), min_ num (8), min_ num(6)) + 1= 3
min_ num (11) = min(min_ num (10), min_ num (9), min_ num(7)) + 1= 4
min_ num (12) = min(min_ num (11), min_ num (10), min_ num(8)) + 1= 3
min_ num (13) = min(min_ num (12), min_ num (11), min_ num(9)) + 1= 4
min_ num (14) = min(min_ num (13), min_ num (12), min_ num(10)) + 1= 4
min_ num (15) = min(min_ num (14), min_ num (13), min_ num(11)) + 1= 5
min_ num (16) = min(min_ num (15), min_ num (14), min_ num(12)) + 1= 4
min_ num (17) = 1
min_ num (18) = min(min_ num (17), min_ num (16), min_ num(14), min_ num(1)) + 1= 2
min_ num (19) = min(min_ num (18), min_ num (17), min_ num(15), min_ num(2)) + 1= 2
min_ num (20) = min(min_ num (19), min_ num (18), min_ num(16), min_ num(3)) + 1= 3
min_ num (21) = 1

min_ num (22) = min(min_ num (21), min_ num (20), min_ num(18), min_ num(5), min_ num(1)) + 1= 2
min_ num (23) = min(min_ num (22), min_ num (21), min_ num(19), min_ num(6), min_ num(2)) + 1= 2
min_ num (24) = min(min_ num (23), min_ num (22), min_ num(20), min_ num(7), min_ num(3)) + 1= 3
min_ num (25) = min(min_ num (24), min_ num (23), min_ num(21), min_ num(8), min_ num(4)) + 1= 2
min_ num (26) = min(min_ num (25), min_ num (24), min_ num(22), min_ num(9), min_ num(5)) + 1= 3
min_ num (27) = min(min_ num (26), min_ num (25), min_ num(23), min_ num(10), min_ num(6)) + 1= 3
min_ num (28) = min(min_ num (27), min_ num (26), min_ num(24), min_ num(11), min_ num(7)) + 1= 4

2c Algorithm (saved_ stamps refers to the length of the array within that index position)
min_ num (1) = 0(original saved_ stamps[1] value) + 1 = 1
min_ num (2) = 0(original saved_ stamps[2] value) + 1 = 1
min_ num (3) = min(saved_ stamps[1] (1), saved_ stamps[2] (1)) + 1 = 2
min_ num (4) = 0(original saved_ stamps[3] value) + 1= 1
min_ num (5) = min(saved_ stamps[4] (1), saved_ stamps[3] (2), saved_ stamps[1] (1)) + 1 = 2
min_ num (6) = min(saved_ stamps[5] (2), saved_ stamps[4] (1), saved_ stamps[2] (1)) + 1 = 2
min_ num (7) = min(saved_ stamps[6] (2), saved_ stamps[5] (2), saved_ stamps[3] (2)) + 1= 3
min_ num (8) = min(saved_ stamps[7] (3), saved_ stamps[6] (2), saved_ stamps[4] (1)) + 1= 2
min_ num (9) = min(saved_ stamps[8] (2), saved_ stamps[7] (3), saved_ stamps[5] (2)) + 1= 3
min_ num (10) = min(saved_ stamps[9] (3), saved_ stamps[8] (2), saved_ stamps[6] (2)) + 1= 3
min_ num (11) = min(saved_ stamps[10] (3), saved_ stamps[9] (3), saved_ stamps[7] (3)) + 1= 4
min_ num (12) = min(saved_ stamps[11] (4), saved_ stamps[10] (3), saved_ stamps[8] (2)) + 1= 3
min_ num (13) = min(saved_ stamps[12] (4), saved_ stamps[11] (4), saved_ stamps[9] (3)) + 1= 4
min_ num (14) = min(saved_ stamps[13] (4), saved_ stamps[12] (3), saved_ stamps[10] (3)) + 1= 4
min_ num (15) = min(saved_ stamps[14] (4), saved_ stamps[13] (4), saved_ stamps[11] (4)) + 1= 5
min_ num (16) = min(saved_ stamps[15] (5), saved_ stamps[14] (4), saved_ stamps[12] (3)) + 1= 4
min_ num (17) = 0(original saved_ stamps[17] value) + 1= 1
min_ num (18) = min(saved_ stamps[17] (1), saved_ stamps[16] (4), saved_ stamps[14] (4), saved_ stamps[1] (1)) + 1= 2
min_ num (19) = min(saved_ stamps[18] (2), saved_ stamps[17] (1), saved_ stamps[15] (5), saved_ stamps[2] (1)) + 1= 2
min_ num (20) = min(saved_ stamps[19] (2), saved_ stamps[18] (2), saved_ stamps[16] (4), saved_ stamps[3] (2)) + 1= 3
min_ num (21) = 0(original saved_ stamps[21] value) + 1= 1
min_ num (22) = min(saved_ stamps[21] (1), saved_ stamps[20] (2), saved_ stamps[18] (2), saved_ stamps[5] (2), saved_ stamps[1] (1)) + 1= 2
min_ num (23) = min(saved_ stamps[22] (2), saved_ stamps[21] (1), saved_ stamps[19] (2), saved_ stamps[6] (2), saved_ stamps[2] (1)) + 1= 2
min_ num (24) = min(saved_ stamps[23] (2), saved_ stamps[22] (2), saved_ stamps[20] (3), saved_ stamps[7] (3), saved_ stamps[3] (2)) + 1= 3
min_ num (25) = min(saved_ stamps[24] (3), saved_ stamps[23] (2), saved_ stamps[21] (1), saved_ stamps[8] (2), saved_ stamps[4] (1)) + 1= 2
min_ num (26) = min(saved_ stamps[25] (2), saved_ stamps[24] (3), saved_ stamps[22] (2), saved_ stamps[9] (3), saved_ stamps[5] (2)) + 1= 3
min_ num (27) = min(saved_ stamps[26] (3), saved_ stamps[25] (2), saved_ stamps[23] (2), saved_ stamps[10] (3), saved_ stamps[6] (2)) + 1= 3
min_ num (28) = min(saved_ stamps[27] (3), saved_ stamps[26] (3), saved_ stamps[24] (3), saved_ stamps[11] (4), saved_ stamps[7] (3)) + 1= 4