CSC236 Summer 2017: A2 Solutions

July 22, 2017

1. (a) Let $T(n) = S(n) + 1$.
   Then we have the following:

$$T(n) = \begin{cases} 2 & \text{if } n = 0 \\ (S(n-1))^2 + 2S(n-1) + 1 & \text{if } n > 0 \end{cases} \tag{1}$$

Look at the second case, we derive the following:

$$\begin{aligned} T(n) &= (S(n-1))^2 + 2S(n-1) + 1 \\ T(n) &= (S(n-1) + 1)^2 \\ T(n) &= (T(n-1))^2 \\ T(n) &= (T(n-1-k))^{2\times 2^k} \\ T(n) &= (T(1))^{2^{n-1}} \\ T(n) &= 2^{2^{n-1}} \end{aligned} \tag{2}$$

Since $T(n) = S(n) + 1$, we know that $S(n) = 2^{2^{n-1}} - 1$.

We will prove the correctness of the closed form with simple induction.

Base case: $n = 0$.
$S(0) = 2^{2^0} = 2^1 = 2$.
Therefore the base case is correct.

Our inductive hypothesis is that $S(n) = 2^{2^{n-1}} - 1$ for some $n \in \mathbb{N}, n \geq 0$.
We want to show that $S(n+1) = 2^{2^n} - 1$.
We can derive this from the recurrence relationship:

$$\begin{aligned} S(n+1) &= (S(n))^2 + 2S(n) \\ S(n+1) &= (2^{2^{n-1}} - 1)^2 + 2(2^{2^{n-1}} - 1) \quad \text{(I.H.)} \\ S(n+1) &= 2^{2^n} - 2(2^{2^{n-1}}) + 1 + 2(2^{2^{n-1}}) - 2 \\ S(n+1) &= 2^{2^n} - 1 \end{aligned} \tag{3}$$

This proves the correctness of our closed form.
Then $\forall n \in \mathbb{N}$, we know that $2^{2^{n-1}} - 1 < 2^{2^{n-1}}$ and $2^{2^{n-1}} - 1 > \frac{1}{2}(2^{2^{n-1}})$. This means that $S(n) \in \Theta(2^{2^n})$.

(b) We will begin by finding $S(n)$ for even n. Let $n = 2k$. By definition we have the following:

$$\begin{aligned}
S(n) &= S(n-2) + 2n - 1 \\
S(n) &= S(n-2i) + 2n + 2(n-2) + \cdots + 2(n-2i+2) - i \\
S(n) &= S(0) + 2n + 2(n-2) + \cdots + 2(2) - k \\
S(n) &= 2\sum_{j=1}^{k} 2j - k \\
S(n) &= 2k(k+1) - k \\
S(n) &= 2k^2 + k
\end{aligned} \tag{4}$$

Now we find $S(n)$ for odd n. Let $n = 2k+1$. By definition we have the following:

$$\begin{aligned}
S(n) &= S(n-2) + 3n \\
S(n) &= S(n-2i) + 3n + 3(n-2) + \cdots + 3(n-2i+2) \\
S(n) &= S(1) + 3n + 3(n-2) + \cdots + 3(3) \\
S(n) &= 1 + 3\sum_{j=1}^{k} (2j+1) \\
S(n) &= 1 + 3k(k+1) + 3k \\
S(n) &= 3k^2 + 6k + 1
\end{aligned} \tag{5}$$

We will prove the correctness of the closed forms with simple induction. We start with the even case again.

Base case: $k = 0$.
$S(0) = S(2k) = 2k^2 + k = 0 + 0 = 0$.
Therefore the base case is correct.

Our inductive hypothesis is that $S(2k) = 2k^2 + k$ for some $k \in \mathbb{N}, n \geq 0$.
We want to show that $S(2(k+1)) = 2(k+1)^2 + k + 1$.
We can derive this from the recurrence relationship:

$$\begin{aligned}
S(2k+2) &= S(2k) + 2(2k+2) - 1 \\
S(2k+2) &= (2k^2 + k) + 2(2k+2) - 1 \quad \text{(I.H.)} \\
S(2k+2) &= 2k^2 + 4k + 2 + k + 1 \\
S(2k+2) &= 2(k^2 + 2k + 1) + k + 1 \\
S(2k+2) &= 2(k+1)^2 + k + 1
\end{aligned} \tag{6}$$

This proves the correctness of our closed form for the even case.

We now prove the correctness of the closed form in the odd case with simple induction.

Base case: $k = 0$.
$S(1) = S(2k+1) = 3k^2 + 6k + 1 = 0 + 0 + 1 = 1$.

Therefore the base case is correct.

Our inductive hypothesis is that $S(2k+1) = 3k^2 + 6k + 1$ for some $k \in \mathbb{N}, n \geq 0$.
We want to show that $S(2(k+1)+1) = 3(k+1)^2 + 6(k+1) + 1$.
We can derive this from the recurrence relationship:

$$\begin{aligned}
S(2k+1+2) &= S(2k+1) + 3(2k+3) \\
S(2k+1+2) &= 3k^2 + 6k + 1 + 3(2k+3) \quad \text{(I.H.)} \\
S(2k+1+2) &= 3k^2 + 6k + 3 + 6k + 1 + 6 \\
S(2k+1+2) &= 3(k^2 + 2k + 1) + 6(k+1) + 1
\end{aligned} \tag{7}$$

This proves the correctness of our closed form for the odd case.

Finally we need to show that $S(n) \in \Theta(n^2)$. Here all the k's are in $\mathbb{N}$.
We know that for $n = 2k$, $S(n) = 2k^2 + k = \frac{n^2}{2} + \frac{n}{2}$.
We also know that for $n = 2k+1$, $S(n) = 3k^2 + 6k + 1 = \frac{3}{4}(4k^2 + 4k + 1) + 3k + \frac{1}{4} = \frac{3}{4}(2k+1)^2 + \frac{3}{2}(2k+1) - \frac{5}{4} = \frac{3}{4}n^2 + \frac{3}{2}n - \frac{5}{4}$.
Notice that for $n > 2$, $\frac{3}{4}n^2 + \frac{3}{2}n - \frac{5}{4} - \frac{n^2}{2} + \frac{n}{2} = \frac{n^2}{4} + n - \frac{5}{4} > 0$.
Furthermore, for $n > 2$, we know that $\frac{3}{4}n^2 + \frac{3}{2}n - \frac{5}{4} < \frac{3}{4}n^2 + n^2 = 2n^2$.
We also point out that for all n, $\frac{n^2}{2} + \frac{n}{2} > \frac{n^2}{2}$.
This means that for sufficiently large $n2$, we have the following relationship:

$$\frac{n^2}{2} \leq \frac{n^2}{2} + \frac{n}{2} \leq S(n) \leq \frac{3}{4}n^2 + \frac{3}{2}n - \frac{5}{4} \leq 2n^2, \tag{8}$$

Therefore $S(n) \in \Theta(n^2)$.

2. In order to prove $S(n) \in \Theta(n \log n)$, we need to show the following:

$$\exists N, c_1, c_2 \in \mathbb{R} \text{ such that } \forall n > N, c_1 n \log n \leq S(n) \leq c_2 n \log n$$

Let N be a number such that $\forall n > N$, the following holds:

(a) $n \log n > 6n - 5$
(b) $\frac{3}{2}n - 5 > 0$
(c) $N > 2^{k_0}$ for which $k_0 > 1$

Since $n \log n$ grows faster than $6n - 5$, we know such an N exists.
Now consider $n > N$, then by hint a, we know that $\exists k \in N$ such that $2^{k-1} \leq n \leq 2^k$.

Furthermore, since S is monotonically non-decreasing, we know the following:

$$\begin{aligned}
S(n) &\geq S(2^{k-1}) \\
&= 2^{k-1}\log 2^{k-1} + 3(2^{k-1}) - 5 \quad \text{(plugging in expression)} \\
&= 2^{k-1}(k-1)\log 2 + \frac{3}{2}(2^k) - 5 \quad \text{(log properties)} \\
&= \frac{k-1}{2k}2^k(k)\log 2 + \frac{3}{2}(2^k) - 5 \quad \text{(arithmetics)} \\
&= \frac{k-1}{2k}2^k\log 2^k + \frac{3}{2}(2^k) - 5 \quad \text{(log properties)} \\
&\geq \frac{k-1}{2k}n\log n + \frac{3}{2}n - 5 \quad \text{(monotonicity of } n\log n) \\
&\geq \frac{k-1}{2k}n\log n \quad (n > N \Rightarrow \frac{3}{2}n - 5 > 0) \\
&= \frac{1}{4}n\log n \quad (k > k_0 > 1 \Rightarrow \frac{k-1}{k} \geq \frac{1}{2}) \\
&= \frac{1}{4}n\log n
\end{aligned} \tag{9}$$

So we can choose $c_1 = \frac{1}{4}$.
In addition, we have the following:

$$\begin{aligned}
S(n) &\leq S(2^k) \\
&= 2^k\log 2^k + 3(2^k) - 5 \quad \text{(plugging in expression)} \\
&= 2^k(k)\log 2 + 6(2^{k-1}) - 5 \quad \text{(log properties)} \\
&= 2\frac{k}{k-1}2^{k-1}(k-1)\log 2 + 6(2^{k-1}) - 5 \quad \text{(arithmetics)} \\
&= 2\frac{k}{k-1}2^{k-1}\log 2^{k-1} + 6(2^{k-1}) - 5 \quad \text{(log properties)} \\
&= 2\frac{k}{k-1}n\log n + 6n - 5 \quad \text{(monotonicity of } n\log n) \\
&\leq 2\frac{k}{k-1}n\log n + n\log n \quad (n > N \Rightarrow n\log n > 6n - 5) \\
&\leq 2(2)n\log n + n\log n \quad (k > k_0 \Rightarrow \frac{k}{k-1} \leq 2) \\
&= 5n\log n
\end{aligned} \tag{10}$$

So we can choose $c_2 = 5$.

3. (a) Let n be the length of the sequence. Then, we present the following algorithm:

```
findMaxConsecutive(A):
maxSum = −∞.
for i = 1 to n
    tempSum = 0;
    for j = i to n
        tempSum = tempSum + A[j]
        if tempSum > maxSum
            maxSum = tempSum
```

return maxSum

It always run $\sum_{i=1}^{n} i$ iterations. Since we know that $\sum_{i=1}^{n} i < n^2$, we know that my algorithm will always have worst case $O(n^2)$ run time.

(b) We have the following recurrence relationship:

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) + n & \text{if } n > 1 \end{cases} \tag{11}$$

In this case, $a = 2$ and $b = 2$.

(c) We have the following Divide and Conquer Algorithm:

findMaxConsecutive(A)
if len(A) = 1
  return $A[1]$
m = $\lfloor \frac{\text{len}(A)}{2} \rfloor$.
find $k_1 \leq m$ such that $\sum_{i=k_1}^{m} A[i]$ is maximized. (Linear search)
find $k_2 \geq m$ such that $\sum_{i=m}^{k_2} A[i]$ is maximized. (Linear search)
leftMax = findMaxConsecutive($A[1 \text{ to } m-1]$) if possible (i.e. $m - 1 > 0$)
rightMax = findMaxConsecutive($A[m+1 \text{ to last element}]$)
return max(leftMax, rightMax, $\sum_{i=k_1}^{k_2} A[i]$)

(d) Based on the recurrence relation, we know that $a = 2$, $b = 2$, and $d = 1$.
Since $a = 2 = b^d$, we know that $T(n) \in \Theta(n \log n)$ by Master Theorem.

4. (a) The property we will leverage can be summarized as follows:

Each way a postage of n cents can be made of 4-cent and 6-cent stamps will either include at least one 4-cent stamp or include no 4-cent stamps.

In order for a postage of n cents to be made of no 4-cent stamps, n would need to be a multiple of 6, this can be easily verified. Furthermore, each way of making the postage of n cents which includes at least one 4-cent stamp can be used to make the postage of $n - 4$ cents by simply removing one of the 4-cent stamps. Then the distinct ways to create a postage of n cents which includes at least one 4-cent stamp is the same as the distinct ways to create a postage of $n - 4$ cents since we can just add a 4-cent stamp and this preserves the uniqueness of each way of making the postage of n cents.

Therefore we derive the following algorithm:

findDistinctWays(A):
if $A = 4$ or $A = 6$
  return 1
return findDistinctWays($A - 4$) + ($\text{mod}(A, 6) = 0$)

(b) **SOLUTION 1**

We derive the following recurrence relation:

$$S(n) = \begin{cases} 0 & \text{if } n < 4, n \neq 0 \\ 1 & \text{if } 4 \leq n \leq 8 \\ 2 & \text{if } n = 10 \\ S(n-4) + S(n-6) - S(n-10) + (\mathrm{mod}(n, 10) = 0) & \text{if } n \geq 4 \end{cases} \tag{12}$$

In this case we know that we can form a postage of 0 cents with 0 4-cent, 0 6-cent, and 0 10-cent stamps. To understand the recurrence relation for $n \geq 4$. For lack of better term, we will treat forming a postage of n cents with i 4-cent, j 6-cent, and j 10-cent stamps as a tuple (i, j, k). For example, we can make 46 with 1 4-cent, 2 6-cent, and 3 10-cent stamp. In this case we can think of our tuple as (1,2,3). We will categorize the tuples into 3 categories - sets containing only 10-cent stamps, sets containing 10-cent and at least one 6-cent stamps, but no 4-cent stamps, and sets containing at least one 4-cent stamp.

Number of tuples using only 10-cent stamps:
These are tuples of the form (0,0,k), where $k \geq 0, k \in \mathbb{N}$.
We can only do this if n is a multiple of 10. This can be computed by $(\mathrm{mod}(n, 10) = 0)$.

Number of tuples using 10-cent stamps, at least one 6-cent stamps, and no 4-cent stamps:
These are tuples of the form (0,j,k), where $k \geq 0, j \geq 1, j, k \in \mathbb{N}$.
We can compute this by looking at the number of tuples of the form (i,j,k) where $i, k \geq 0, j \geq 1, i, j, k \in \mathbb{N}$, we call this S_1 and the number of tuples of the form (i,j,k) where $k \geq 0, i, j \geq 1, i, j, k \in \mathbb{N}$, we call this S_2. Formally, we can think of S_1 as the number of tuples that require at least 1 6-cent stamps, but places no restriction on the number of 4-cent or 10-cent stamps used. Also, we can think of S_2 as the number of tuples that require at least 1 6-cent stamp and at least 1 4-cent stamp, but places no restriction on the number of 10-cent stamps used. By taking their difference, we compute the number of tuples that require at least 1 6-cent stamp and 0 4-cent stamps, but places no restriction on the number of 10-cent stamps used. Since we know that each tuple contributing to S_1 contains at least 1 6-cent stamps, these tuples can be formed by looking at the distinct ways to form a postage of $n - 6$ cents, and then adding a 6-cent stamp to them. Therefore $S_1 = S(n - 6)$. Similarly we can compute S_2 by looking at the distinct ways to form a postage of $n - 10$ cents, and then adding a 6-cent stamp and a 4-cent stamp to them. Therefore $S_2 = S(n - 10)$. Then the number of tuples that use 10-cent stamps, at least one 6-cent stamps and no 4-cent stamps to form a postage of n cents is $S(n - 6) - S(n - 10)$.

Number of tuples using 10-cent stamps, at least one 4-cent stamps:
These are tuples of the form (i,j,k), where $j, k \geq 0, i \geq 1, i, j, k \in \mathbb{N}$.
We can compute this easily by looking at the ways to form a postage of $n - 4$ cents and then adding a 4-cent stamp to each. Therefore this is computed by $S(n - 4)$.

Putting the three observations together yields our recurrence relation.

SOLUTION 2

We use the recurrence in part a to compute $S(n)$.

Recall that $T(n)$ yields the number of tuples which use only 4-cent and 6-cent stamps to make a postage of n cents.

Then we have the following recurrence relation:

$$S(n) = \begin{cases} 0 & \text{if } n < 4, n \neq 0 \\ 1 & \text{if } 4 \leq n \leq 8 \\ 2 & \text{if } n = 10 \\ S(n-10) + T(n) & \text{if } n \geq 4 \end{cases} \tag{13}$$

Effectively $S(n-10)$ tells us the number of tuples using at least 1 10-cent stamp. $T(n)$ tells use the number of tuples using no 10-cent stamps. Their sum together provides the number of tuples that can make a postage of n cents.

(c) **PROOF FOR SOLUTION 1**

We will prove that $S(n)$ is non-decreasing by strong induction:

Base case: $k = 4, 6, \ldots, 26$.

These base cases can be verified with brute force. It will become apparent why we need to verify $n = 26$ as a base case.

Our inductive hypothesis is that $S(k) - S(k-2) \geq 0 \ \forall k \in \mathbb{N}, n \geq k \geq 4$ for even $n > 26$.

We want to show that $S(n+2) - S(n) \geq 0$.

We begin by applying the recurrence relation to $S(n+2) - S(n)$:

$$\begin{aligned} S(n+2) - S(n) &= S(n-2) + S(n-4) - S(n-8) + (\text{mod}(k+2, 10) = 0) \\ &\quad - S(n-6) - S(n-4) + S(n-10) - (\text{mod}(k, 10) = 0) \\ &\geq S(n-2) - S(n-6) - S(n-8) + S(n-10) \\ &\quad - (\text{mod}(k, 10) = 0) \end{aligned} \tag{14}$$

We know that $S(n-2) = S(n-6) + S(n-8) - S(n-12) + (\text{mod}(n-2, 10) = 0)$ due to the recurrence relation.

$$\begin{aligned} S(n+2) - S(n) &\geq S(n-2) - [S(n-6) + S(n-8) - S(n-12) + (\text{mod}(k-2, 10) = 0)] \\ &\quad + S(n-10) - S(n-12) + (\text{mod}(k-2, 10) = 0) - (\text{mod}(k, 10) = 0) \\ &\geq S(n-2) - S(n-2) + S(n-10) - S(n-12) + (\text{mod}(k-2, 10) = 0) \\ &\quad - (\text{mod}(k, 10) = 0) \\ &\geq S(n-10) - S(n-12) + (\text{mod}(k-2, 10) = 0) - (\text{mod}(k, 10) = 0) \\ &\geq S(n-10) - S(n-12) + (\text{mod}(k-12, 10) = 0) - (\text{mod}(k-10, 10) = 0) \end{aligned} \tag{15}$$

We know that $S(n-10) - (\text{mod}(k-10, 10) = 0) = S(n-14) + S(n-16) - S(n-20)$ by the recurrence relation. Furthermore, $S(n-12) - (\text{mod}(k-12, 10) = S(n-16) + S(n-$

18) − $S(n-22)$.

$$\begin{aligned} S(n+2)-S(n) &\geq [S(n-10)-(\text{mod}(k-10,10)=0)]-[S(n-12)-(\text{mod}(k-12,10)=0)] \\ &\geq [S(n-14)+S(n-16)-S(n-20)]-[S(n-16)+S(n-18)-S(n-22)] \\ &\geq S(n-14)-S(n-20)-S(n-18)+S(n-22) \end{aligned} \tag{16}$$

We know that $S(n-14) = S(n-18) + S(n-20) - S(n-24) + (\text{mod}(n-14, 10) = 0)$ due to the recurrence relation.

$$\begin{aligned} S(n+2)-S(n) &\geq S(n-14)-[S(n-20)+S(n-18)-S(n-24)+(\text{mod}(n-14,10)=0)] \\ &\qquad +S(n-22)-S(n-24)+(\text{mod}(n-14,10)=0) \\ &\geq S(n-14)-S(n-14)+S(n-22)-S(n-24)+(\text{mod}(n-14,10)=0) \\ &\geq S(n-22)-S(n-24) \\ &\geq 0 \quad (\text{Since } n\geq 28 \text{ we know that } n-24\geq 4 \text{ so we can apply the I.H.}) \end{aligned} \tag{17}$$

Therefore, $S(n)$ is non-decreasing.

PROOF FOR SOLUTION 2

Again, we will prove that $S(n)$ is non-decreasing by strong induction:

Base case: $k = 4, 6, \ldots, 22$.

These base cases can be verified with brute force. It will become apparent why we need to verify $n = 22$ as a base case.

Our inductive hypothesis is that $S(k) - S(k-2) \geq 0 \ \forall k \in \mathbb{N}, n \geq k \geq 4$ for even $n > 22$.

We want to show that $S(n+2) - S(n) \geq 0$.

We begin by applying the recurrence relation to $S(n+2) - S(n)$:

$$\begin{aligned} S(n+2)-S(n) &= T(n+2)-T(n)+S(n-8)-S(n-10) \\ &= T(n+2)-T(n)+[S(n-18)-S(n-20)+T(n-8)-T(n-10)] \\ &= S(n-18)-S(n-20)+T(n+2)-T(n)+T(n-8)-T(n-10) \end{aligned} \tag{18}$$

We point out that $-T(n) + T(n-8) = -T(n-4) - (\text{mod}(n, 6) = 0) + T(n-8) = -(\text{mod}(n-4, 6) = 0) - (\text{mod}(n, 6) = 0)$.

Also, $T(n+2)-T(n-10) = T(n-2)+(\text{mod}(n+2, 6) = 0)-T(n-10) = T(n-6)+(\text{mod}(n-2, 6) = 0) + (\text{mod}(n+2, 6) = 0) - T(n-10) = (\text{mod}(n-6, 6) = 0) + (\text{mod}(n-2, 6) = 0) + (\text{mod}(n+2, 6) = 0)$.

We can apply these two equations to Equation 18 for the following:

$$\begin{aligned} S(n+2)-S(n) &= S(n-18)-S(n-20)+(\text{mod}(n-6,6)=0)+(\text{mod}(n-2,6)=0) \\ &\qquad +(\text{mod}(n+2,6)=0)-(\text{mod}(n-4,6)=0)-(\text{mod}(n,6)=0) \\ &= S(n-18)-S(n-20)+(\text{mod}(n-2,6)=0) \\ &\geq 0 \quad (\text{Since } n\geq 24 \text{ we know that } n-20\geq 4 \text{ so we can apply the I.H.}) \end{aligned} \tag{19}$$

Therefore, $S(n)$ is non-decreasing.

5. (a) ICP(x, y, n):

   minDistance $= (x_0 - x_1)^2 + (y_0 - y_1)^2$
   pairs = (0,1)
   $p = 0$
   while $p < n - 1$
       $q = i + 1$
       while $q < n$
           if $(x_p - x_q)^2 + (y_p - y_q)^2 <$ minDistance
               minDistance $= (x_p - x_q)^2 + (y_p - y_q)^2$
               pairs $= (p, q)$
           $q = q + 1$
       $p = p + 1$
   return $([x_{\text{pairs}}(0), y_{\text{pairs}}(0)], [x_{\text{pairs}}(q), y_{\text{pairs}}(q)])$

   (b) Since we have always have $1 + 2 + \cdots + n$ iterations, we have $O(n^2)$ iterations. The operations in each iteration takes constant time. Therefore, the complexity class of our algorithm is $O(n^2)$.

   (c) Let $p_j^{(i)}$ be the value of p, $q_j^{(i)}$ be the value of q, minDistance$_j^{(i)}$ be the value of minDistance and pairs$_j^{(i)}$ be the value of pairs at outer loop iteration i and inner loop iteration j. Then, we have the following loop invariants:

   Outer loop invariant:

   Suppose the outer loop executed i times. Then $P(i)$: at the end of the i^{th} iteration, $p = i$ and pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\Big\{(a, b) | \bigcup_{k=1}^{i} \{0, 1, \ldots, k - 1\} \times \{k, k + 1, \ldots n - 1\}\Big\}$$

   Inner loop invariant:

   Suppose that before the inner loop executes, that $p = i$ and $P(i)$. Now suppose that the inner loop executes at least j times, then $Q(i, j)$: at the end of the j^{th} iteration, $q = i + j + 1$ and pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\Big\{(a, b) | \bigcup_{k=1}^{i} \{0, 1, \ldots, k - 1\} \times \{k, k + 1, \ldots n - 1\}\Big\} \cup \Big\{(a, b) | \{i\} \times \{i + 1, i + 2, \ldots i + j\}\Big\}$$

   We begin by showing the inner loop invariant holds for each iteration.

   Base case: $Q(i, 0)$ holds.

   By the initialization of q, we know that $q = p + 0 + 1$. Furthermore, since $P(i)$ holds, we know that pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\Big\{(a, b) | \bigcup_{k=1}^{i} \{0, 1, \ldots, k - 1\} \times \{k, k + 1, \ldots n - 1\}\Big\}.$$

Since $j = 0$, we know that

$$\left\{(a,b)|\bigcup_{k=1}^{i}\{i\} \times \{i+1, i+2, \dots i+j\}\right\} = \emptyset.$$

This means that $Q(i, 0)$ holds.

Our inductive hypothesis is that $Q(i, j)$ holds.
We needs to show that $Q(i, j + 1)$ holds.
Suppose the loop executes $j + 1$ times. Then we know that $q = i + j + 1 < n$, otherwise we would have terminated.
Then there are two things that can happen in the inner loop:

i. $(x_p - x_q)^2 + (y_p - y_q)^2 \geq$ minDistance. In this case pair and minDistance do not change. Since $Q(i, j)$ holds, we know that pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\left\{(a,b)|\bigcup_{k=1}^{i}\{0,1,\dots,k-1\}\times\{k,k+1,\dots n-1\}\right\}\cup\left\{(a,b)|\{i\}\times\{i+1,i+2,\dots i+j\}\right\}.$$

$(x_p - x_q)^2 + (y_p - y_q)^2 \geq$ minDistance, we know in fact that pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\left\{(a,b)|\bigcup_{k=1}^{i}\{0,1,\dots,k-1\}\times\{k,k+1,\dots n-1\}\right\}\cup\left\{(a,b)|i\bigcup\{i\}\times\{i+1,i+2,\dots i+j+1\}\right\}.$$

ii. $(x_p - x_q)^2 + (y_p - y_q)^2 <$ minDistance. In this case, we change pair to be the current indices, since the current pair have a smaller distance than any of the indices in the set:

$$\left\{(a,b)|\bigcup_{k=1}^{i}\{0,1,\dots,k-1\}\times\{k,k+1,\dots n-1\}\right\}\cup\left\{(a,b)|\{i\}\times\{i+1,i+2,\dots i+j\}\right\}.$$

This means that pair still stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\left\{(a,b)|\bigcup_{k=1}^{i}\{0,1,\dots,k-1\}\times\{k,k+1,\dots n-1\}\right\}\cup\left\{(a,b)|\{i\}\times\{i+1,i+2,\dots i+j+1\}\right\}.$$

In either case, we set $q_{\text{new}} = q_{\text{old}} + 1 = i + j + 1$.
Therefore $Q(i, j)$ holds.

Now we show that the outer loop invariant holds for each iteration:
Base case: $P(0)$ holds.
By the initialization of p, we know that $p = 0$.

Our inductive hypothesis is that $P(i)$ holds.
We needs to show that $P(i + 1)$ holds.
Suppose the outer loop executes $i+1$ times. Then we know that $p = i < n-1$, otherwise

we would have terminated. Then q is initialized to $p+1 < n$. This means that the inner loop will execute. At the end of the outer loop iteration $i+1$, the inner loop must also have finished executed. Since $p = i$, the inner loop executed at least $n-i-1$ times.This means that $Q(i, n-i-1)$ holds. This means that pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\Big\{(a,b)|\bigcup_{k=1}^{i}\{0,1,\ldots,k-1\}\times\{k,k+1,\ldots n-1\}\Big\}\cup\Big\{(a,b)|\{i\}\times\{i+1,i+2,\ldots i+j\}\Big\}.$$

Then we know that $i+j+1 = n$ which results in the following:

$$\Big\{(a,b)|\bigcup_{k=1}^{i}\{0,1,\ldots,k-1\}\times\{k,k+1,\ldots n-1\}\Big\}\cup\Big\{(a,b)|\{i\}\times\{i+1,i+2,\ldots n-1\}\Big\}.$$

We can combine the two sets to yield:

$$\Big\{(a,b)|\bigcup_{k=1}^{i+1}\{0,1,\ldots,k-1\}\times\{k,k+1,\ldots n-1\}\Big\}.$$

Finally we increment p by 1.
Therefore $P(i+1)$ holds.
We have shown that the inner and outer loop invariants hold.

Now we need to prove partial correctness.
Suppose that the precondition of the algorithm holds and that it terminates. Then it returns a pair of points that have the minimal distance.

Since $n \geq 2$ we know that the outer loop executes at least once. Since the algorithm terminates, we know that $p \geq n-1$ on termination. This means that the other loop executes $n-1$ times. By the outer loop invariant we know that $P(n-1)$ holds. Since we know that $p = n-1$ at the end of iteration $n-1$, this means that the outer loop will not execute again, and the indices stored in pair at the end of iteration $n-1$ will be used to decide the pair of points returned.
By $P(n-1)$ we know that pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\Big\{(a,b)|\bigcup_{k=1}^{n-1}\{0,1,\ldots,k-1\}\times\{k,k+1,\ldots n-1\}\Big\}.$$

By symmetry we also know that pair stores the indices $[x, y]$ such that there is no smaller distance among the following indices:

$$\Big\{(a,b)|\bigcup_{k=1}^{n-1}\{k,k+1,\ldots n-1\}\times\{0,1,\ldots,k-1\}\Big\}.$$

This is an exhaustive comparison of the pairs of points. So pair stores the pair of indices whose points have the minimal distance among the given points.

Finally we need to show that the algorithm terminates.
First we will show that if the inner loop executes, it will terminates.
Let q_j be the value of q before iteration j. We want to show that the sequence $n - q_j$ is decreasing. We know that by initialization, $q_1 = p + 1 < n$ since the inner loop executes. So we have $n - q_1 > 0$. Suppose the inner loop has finishing its j^{th} iteration and is about to execute iteration $j + 1$. Then by $Q(i, j)$, we know that $q_{j+1} = q_j + 1$. Furthermore we know that $q_{j+1} < n$ since the loop is executing again. This yields:

$$0 < n - q_{j+1} = n - q_j - 1 < n - q_j.$$

Now we need to show that if the outer loop executes, it will terminates.
Let p_i be the value of p before iteration i. Since the inner loop will always terminate, we just need to show that the outer loop will terminate as well. We want to show that the sequence $n - p_i$ is decreasing. We know that by initialization, $p_1 = 0 < n$ since the inner loop executes. So we have $n - q_1 > 0$. Suppose the inner loop has finishing its i^{th} iteration and is about to execute iteration $i + 1$. Then by $P(i)$, we know that $p_{i+1} = p_i + 1$. Furthermore we know that $p_{i+1} < n - 1$ since the loop is executing again. This yields:

$$0 < n - p_{i+1} = n - p_i - 1 < n - p_i.$$

This completes the proof.