Homework 6

余北辰 519030910245

a.

Let C_0 = 1 penni = 1 cent, C_1 = 1 nickel = 5 cents, C_2 = 1 dime = 10 cents, C_3 = 1 quarter = 25 cents.
Let m_i = the number of C_i needed.
If we want making change for n cents:

```
int i = 3;
while (i--)
{
    m_i = n / C_i;
    n %= C_i;
}
```

Then we get the number of m_0, m_1, m_2, and m_3.

Proof of correctness:

5 C_0 can get a C_1, so there is 4 C_0 at most. Similarly, we can proof that there are 1 C_1 and 2 C_2 at most. As 1 C_1 and 2 C_2 gets 1 C_3, 1 C_1 and 2 C_2 cannot exist together.
First, we proof that the number of C_3 must be m_3. We assume that there is a better algorithm and the number of C_3 is p_3, because we use the greedy algorithm, $m_3 >= p_3$. But if $m_3 < p_3$, there is at least 25 cents must use C_0, C_1 and C_2 to get. However, it is impossible to get 25 cents with 4 C_0, 1 C_1 and 2 C_2 while 1 C_1 and 2 C_2 cannot exist together. So we proof that $m_3 = p_3$. Using the same method, we can proof that $m_2 = p_2$, $m_1 = p_1$ and $m_0 = p_0$. That is to say that the greedy algorithm is right.

b.

If using the greedy algorithm, $n = m_0c^0 + m_1c^1 + \ldots + m_kc^k$ and we assume that there is a better algorithm, $n = n_0c^0 + n_1c^1 + \ldots + n_kc^k$.

First, we proof that $n_k = m_k$. As it is a greedy algorithm, $m_k >= n_k$. If $n_k < m_k$, then $n_k <= m_k – 1$.
It is absolutely that $n_0, n_1, \ldots, n_{k-1} < c$ as if there is c coins they can be changed to a higher denomination coin. So $n_0c^0 + n_1c^1 + \ldots + n_{k-1}c^{k-1} <= (c - 1)c^0 + (c - 1)c^1 + \ldots + (c - 1)c^{k-1} = c^k – 1 < c^k$. So $n = n_0c^0 + n_1c^1 + \ldots + n_kc^k < c^k + n_kc^k <= m_kc^k < n$, it is absolutely wrong. So $n_k = m_k$. Using the same method, we can proof that for every $i <= k$, $n_i = m_i$. That is to say that the greedy algorithm is right.

c.

We want to get n = 30 cents with C_0 = 1 cent, C_1 = 10 cents, C_2 = 25 cents.
With the greedy algorithm, $n = 1C_2 + 0C_1 + 5C_0$, 6 coins in total.
But in fact $n = 0C_2 + 3C_1 + 0C_0$, just 3 coins in total.